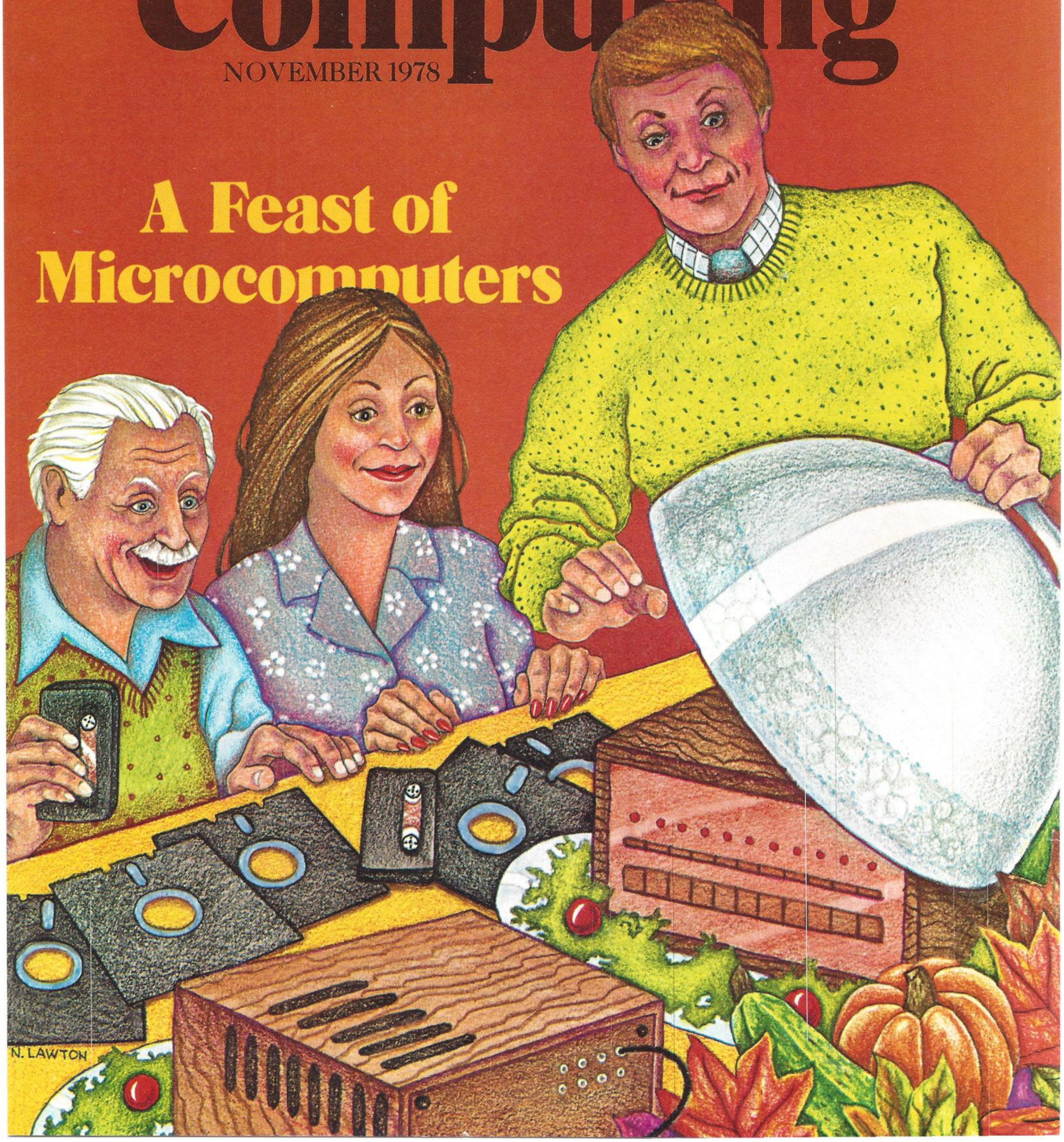
Investment · Dealing with Adversaries · Accounting · Model Rockets

Personal Computing

$2.00

NOVEMBER 1978

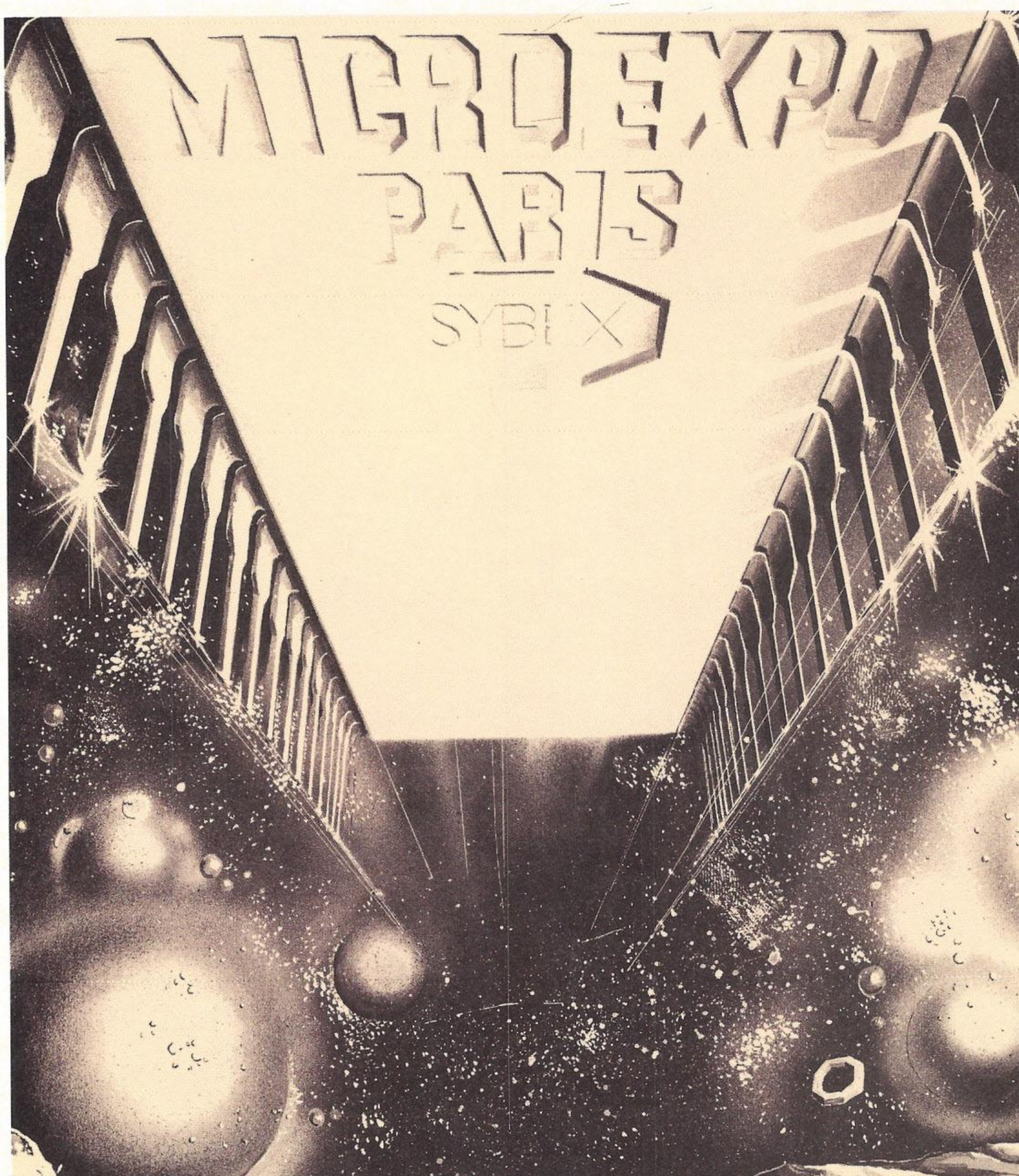
A Feast of Microcomputers



MICRO/EXPO 79
PARIS

MAY 15 - 17

CENTRE INTERNATIONAL DE PARIS
4TH ANNUAL MICROCOMPUTER SHOW



SYBEX

U.S.A.
SYBEX, Inc.
2020 Milvia St., Berkeley, CA 94704
Tel: 415/848-8233 Tlx: 336311

EUROPE
SYBEX EUROPE
313 rue Lecourbe, 75015-Paris, France
Tel: (1) 828 25 02 Tlx: 200858

Totally Integrated, Entirely Self-Contained

THE PET™

PERSONAL COMPUTER

**With technology so advanced,
Concept so remarkable,
Operation so utterly simple,
Cost so incredibly low.
The PET has given rise to a brand new era...
The Age of the Personal Computer**

HIGH SPEED PRINTER ACCESSORY

Immediate Delivery

FEATURING AN IEEE-488 BUS

The PET, unquestionably, has become the standard for the personal computer industry. As such, consumer and business publications have lauded its discovery. It comes out of the box, plugs into the wall, and is ready to use.

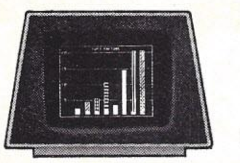
IN A LEAGUE WITH IBM, HP AND WANG MINICOMPUTERS

The fully integrated PET is a minicomputer and should not be confused with game products that hook up to household T.V.'s. **What sets it apart from other computers is price. While others cost from $11,000 to $20,000 and more, THE PET, with similar power, is only $795.00.**

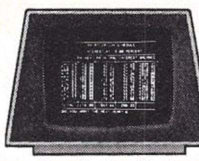
It is of sufficient size and speed to accomplish many of the routine computational tasks facing engineers and scientists. By combining an extended BASIC interpreter with one of the fastest microprocessors available, Commodore has created an incredibly capable, yet inexpensive, computer.

POWERFUL 6502 HARDWARE:

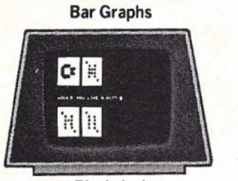
The mind-boggling PET has all necessary functional elements packaged in a single cabinet. Its CPU is a MOS Technology 6502 8 bit microprocessor which can address 65,536 bytes of memory directly. It has 56 instructions and 13 different addressing modes. Depending upon address mode, instructions are executed in 2 to 7 microseconds. The unit has 9216 bytes of random access memory (RAM) and 16,384 bytes of read only memory (ROM). 1K of the RAM and 2K ROM are used for video display. 1000 characters, organized in 25 rows of 40 columns can be displayed at a time. An 8 x 8 dot matrix delivers superb CPU readability. The character set consists of 158 upper and lower case letters, numerals, punctuation symbols and a variety of graphics characters.



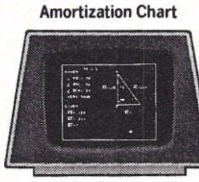
Bar Graphs



Amortization Chart



Black Jack



Teaching Trigonometry

All characters can be displayed in normal mode -white on black, or reverse - black on white. Since the display is part of main memory, it can be accessed directly by the CPU. Flexibility is enhanced by the full screen editor. Input is either from the 10-key numeric keyboard, with its full cursor control, or the built-in cassette tape drive. All symbols used in BASIC statements are available in the unshifted mode for added operating speed. The cassette can be used to read and write programs and data at 50 bytes per second. A second cassette port, 8 bit parallel I/O port and IEEE-488 bus port are also provided. The entire system weighs 38 lbs. and measures approximately 15½" high x 17¼" wide x 18½" deep.

The PET is completely silent in its operation. No special installation is necessary because it operates on ordinary 110V household current at 50-60 Hz.

INCREDIBLE FIRMWARE FLEXIBILITY

The firmware's three primary components include the operating system, full screen editor, and BASIC interpreter. Together these programs use 14 K bytes of read only memory.

DIRECT ACCESSORY PLUG-IN

The unit's operating system permits you to easily load and save programs. One of its exclusive features is complete support for the **IEEE-488 instrument bus,** which is occasionally referred to as an HP-IB bus. This bus provides superb flexibility. Through it, you can connect up to 12 devices to the PET, at the same time. This universal port permits direct connection of the PET Printer, in addition to a wide variety of measurement and control instruments such as counters, timers, spectrum analyzers, digital voltmeters and printer plotters, from HP, Phillips, Fluke and Textronix. Many academic, industrial and governmental labs are already using PETS as laboratory automation control computers.

With the PET's full screen editor, an entire program, a selected section, or a statement can be listed. The cursor can be moved as needed on the screen, and characters can be changed, inserted or deleted to modify any program statement. Statements can be copied or moved by changing their line number. New statements can be added, or old deleted, as desired.

COMPLETE COMPUTER PERFORMANCE

The BASIC interpreter enables a superset of the original BASIC. Among its major features are floating point, integer and string arrays of limited dimensionality, dynamic string handling, 5 byte floating point number representation (for approximately 8 decimal digit accuracy), PEEK and POKE commands for direct memory access, two character variable names, full support for IEEE-488 bus devices, program chaining with full data retention between overlays, built-in mathematical (ABS, ATN, COS, EXP, INT, LOG, RND, SGN, SIN, SQR, TAN) and string (ASC, CHR, LEFT, LEN, MID, RIGHT, STR, VAL) functions; user-definable functions; multi-statement lines; real-time clock; support for machine language subroutines; both character and line input capability.

HIGH SPEED PET PRINTER

This powerful word processor prints hardcopies, invoices, computer correspondence. Faster than an IBM Selectric, The PET Printer delivers 60 characters per second at a sustained rate -- with upper and lower case capability. Characters are one-eighth inch tall and are printed in a large 8 x 7 dot matrix. The printer uses a standard 8½" wide paper roll. And, it is only $695.00.

PERIPHERAL SECOND CASSETTE

This optional component expands storage, increases flexibility, and is just $99.95.

SOFTWARE

Application programs for the PET can of course be written by the user, and most special purpose programs will be so written. A growing library of programs is already available.

PROGRAMS AT $29.95 EACH:

☐ **Basic Math Package:**
Matrix operations-entry, editing, addition, subtraction, multiplication, inversion, determinants, solution of simultaneous equations.
Vector operations-entry, editing, addition, subtraction, dot and cross products, length, angle between vectors.
Plane and spherical trigonometry-calculate sides and angles of triangles, areas of regular polygons.
Numerical integration and differentiation-interactive entry, editing and display of data; Simpson's rule, trapezoidal method and Gaussian quadrature integration algorithms; differentiation allows arbitrary interval size.

☐ **Basic Statistics Package:**
General statistics-calculates mean, median, standard deviation, skewness, kurtosis, frequencies, range, and variance of a data set.
T-tests-to determine the significance of a difference between the means of independent groups, matched groups, mean gains of paired groups, known and observed means and means of sample proportions.
Chi-square test for independence between two statistical processes.
Correlation and regression-calculates Pearson product moment correlation coefficient and the regression line.

PROGRAMS AT $24.95 EACH:

☐ **Basic Investment Analysis:**
Loans, annuities, return on regular and irregular sequences of payments, calendar calculations.
☐ **Stock Portfolio Recordkeeping and Analysis:**
Keeps track of buys, sells, and dividends. Calculates current value, rates of return.
☐ **Checkbook Recordkeeping and Analysis:**
Keeps track of checks and deposits. Analyzes expenses by date and type.

PROGRAMS AT $14.95 EACH:

☐ **Mortgage Analysis**
☐ **Diet Planner and Biorhythm**
☐ **Basic Basic**-*by Lodewyck and James*

GAME PROGRAMS ARE $9.95 EACH:

☐ **Black Jack** ☐ **Draw Poker** ☐ **Galaxy Games**
☐ **Space Flight** ☐ **Target Bong, Off-The-Wall**
☐ **Lunar Lander, Wumpus, Rotate, Tic-Tac-Toe**
☐ **Osero, Reverse** ☐ **Spacetrek** ☐ **Kingdom**

FREE ORIENTATION PACKAGE

Your PET comes complete with an introductory cassette and an easy-to-follow instruction manual.

SERVICE WORLDWIDE

Because your PET is self-contained and compact, professional factory service is never far away. If major service is required, the unit can simply be returned by UPS to an authorized Commodore PET clinic.

To order your PET send check or money order for $795.00 plus $20.00 for shipping and insurance. To order the PET Printer, add $695.00 plus $12.00 for shipping and insurance. The Second Cassette is $99.95. No shipping and insurance charges are required for the second cassette or programs when ordered with your PET. Credit card orders are invited to call our toll free number below. Orders will be accepted on our Telex, No. 25-5268.

Use THE PET for 30 days with no obligation. If, for any reason, you are not satisfied, return it for a prompt and courteous refund.

**ORDER DIRECT
CREDIT CARD ORDERS CALL TOLL FREE**

800-323-2272

**ILLINOIS RESIDENTS CALL: 312-595-0461
TELEX ORDERS: 25-5268**

**790 MAPLE LANE DEPT. PC-11
BENSENVILLE, ILLINOIS 60106**

Contemporary Marketing Inc.

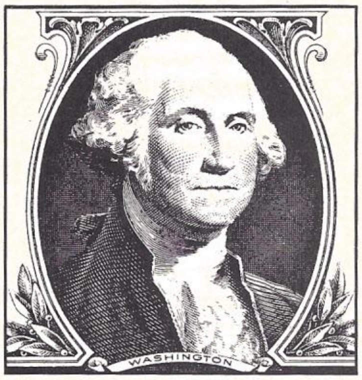
©CMI 1978214

CIRCLE 4

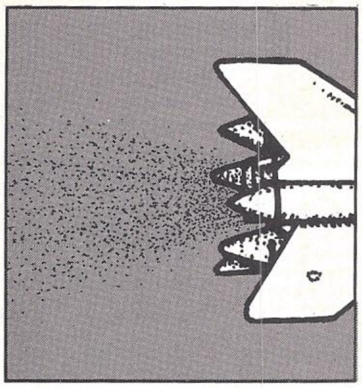
Personal Computing

NOVEMBER 1978 VOLUME II, NO. 11

Page 30



Page 42



Page 48

Cover Illustration by Nancy Lawton

DEPARTMENTS

Personal Computing

NOVEMBER 1978 Vol. II, No. 11

Advertising Sales. Northeast: George Palken, Bill Mahan, 1050 Commonwealth Ave., Boston, MA 02215; (617) 232-5470 • **Mid-Atlantic:** Arthur Daks, Benwill Publishing Corp., 104 So. Central Ave., Valley Stream, NY 11580; (516) 872-9550 • **Mid-West:** Hank Bean, 2633 Hillside Lane, Evanston, IL 60201; (312) 475-7173 • **Northwest:** Ralph Petersen, 1380 Oak Creek Drive, Palo Alto, CA 94304; (415) 328-4392 • **Southwest:** Yuri Spiro, Carol Stagg, Benwill Publishing Corp., 924 Westwood Blvd., Los Angeles, CA 90024; (213) 478-3017 • **Japan:** Hiro H. Irie, International Business Corp., 11-8, Narita-Higashi 1-chome, Suginami-ku, Tokyo 166; Phone (03) 311-1746

Subscription rates. U.S.: 1 year (12 issues) $14; 2 years (24 issues) $26; 3 years (36 issues) $38. Canada & Mexico: add $4/year for surface mail, $8/year for airmail. In Japan/Asia: Personal Computing, c/o CQ Publishing Co., 14-2 Sugamo 1-chome, Toshima-ku, Tokyo 170. Japan. All other countries: add $8/year for surface mail, $36/year for airmail. Send subscription orders and changes of address to: Circulation, Personal Computing, 1050 Commonwealth Ave., Boston, MA 02215. **Back issues.** U.S.: $3. Canada & Mexico: $4. All other countries: $6.

Publisher. Published monthly by Benwill Publishing Corp.: Harold G. Buchbinder, Chairman of the Board; George Palken, President; Esther Shershow, Treasurer. Executive, Editorial and Subscription Offices: 1050 Commonwealth Ave., Boston, MA 02215. Controlled Circulation postage paid at Long Prairie MN. Membership in Audit Bureau of Circulation pending.

Marvelous Mistake

Gentlemen:

I've just been enjoying "The Marvelous Micro Mentalist" by Tim Purinton, August *PC.*

He has sure come a long way since the early efforts reported in your pages in '77.

But like the rest of us, he's still not free from bugs. Notice that in the list of cards dealt by his program, the second time through the deck produces the same series that came out on the first pass. This results from the error at line 6005, which routes us back to line 15. Purinton acknowledges that line 15 should be deleted, since that same work is now done at line 19. To send the program back to 15 now leads to re-seeding the randomizer at line 18 with the same - NX – so of course the run of cards is repeated. Let line 6005 read GOTO 19 and the problem is averted.

N.B. Winkless Jr.
No. Hollywood, CA

Games People Play

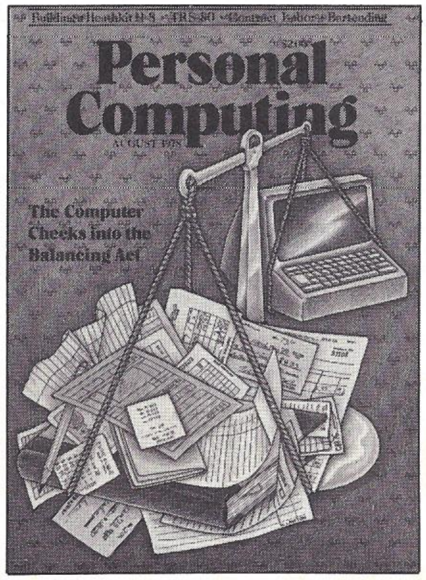
Dear Editors:

Can anyone tell me how two or more persons all using their own TRS-80 can play games over the phone lines so each person's video shows the same thing and reacts identically? Please specify how to make or where to buy any additional hardware that may be required.

Also, it's really frustrating to want to use a Level I program and can't because you have Level II. In most cases, I don't know how to re-do my Level I tapes so they'll run on Level II. I've got 16K Level II and most of my Level I tapes still come back with "program too long" after using the conversion tape on them.

Can anyone tell me how to easily change my Level II back to Level I and vice versa or know of anyone who has developed an inexpensive device to do this without having to bother with Level I to Level II tape conversions?

Sharon Jackson
Fenton, MO



Editor's note: Right now the only way two or more people can play games over telephone lines is via a modem which links the signals from the micro to the phone itself. However, if more than two people are linked together on the phone network, one and only one is the master control and the others are slaves. Two people using originate-answer modems can carry on two-way communications while both see identical CRT displays. Radio Shack plans to announce a modem for their TRS-80 soon. It should accommodate your requirements rather well.

You must be very careful when converting programs from Level I to Level II to set the volume controls correctly. Level I input is at 250 baud and requires a volume setting of 8 to 10; Level II input is at 500 baud and requires a volume setting of 4 to 6.

If you make an error inputting from tape, your TRS-80 responds with a "program too long" advisory, even though the program will actually fit comfortably into your 16K. So checking your volume settings should solve your problem. *– G.W.D.*

Fishing for Bugs

Editors:

I recently purchased an Altair 88-DCDD disk for my computer. I therefore read "Fishing the MOD Way" (April '78) with great eagerness. Not only did it represent an example of modulus arithmetic, but it also contained a well-defined example of the use of random files, with which I had no real experience.

I therefore loaded it onto my disk, and it worked fine as long as I didn't get out of the entry routine. The first printout was fine, but all printouts of those same files *after* leaving the entry routine returned garbage until it reached the point where the most recent entry routine was entered, when it entered and returned perfectly.

After much searching and some help from a friend, I found the trouble. It is in the "TRANSACT" random file created by the program, or more specifically the method of moving from the program's entries to the buffer of that file. Since each subrecord uses only part of a complete record, the buffer is only partly filled by each subrecord. That's fine as long as the rest of the buffer contains what you want it to contain (same program run), but if there is a time difference between the program run and when the buffer was constructed, the only part of the buffer which will be accurately filled is that part which corresponds to the program run. In short, it is necessary to GET the buffer filled with the full record each time you attempt to PUT some information into a subrecord.

The program may be fixed by adding:

```
545 GET #1, J
595 PUT #1, J
```

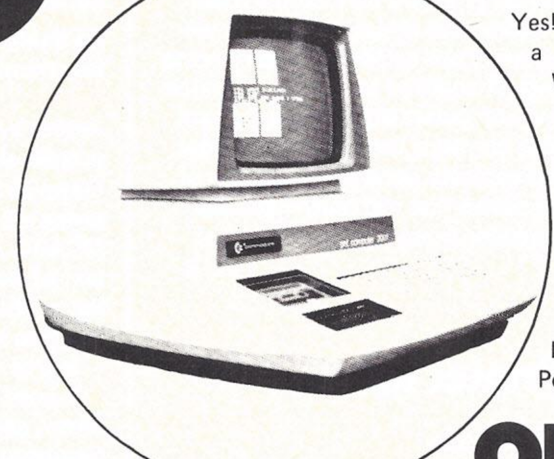
The "LAST TR$" file doesn't need chaining since the same subrecord is used each time. This can be used as the first subrecord in "TRANSACT" if someone doesn't wish to use an entire file for only 2 bytes worth of information.

T.J. Simpson
Sacramento, CA

Author's note: Mr . Simpson is correct, but he will also have to change the GOSUB reference numbers, specifically in lines 300, and 860. The program was merely an illustrating vehicle for entirely different points made in the article. But I am glad that he found it useful. *– O.E. Dial*

PET 2001

IN STOCK!!!
AVAILABLE FOR IMMEDIATE DELIVERY!!!



Yes! — It's hard to believe. The PET-2001, a full-fledged computer by Commodore. What do you get? Full ASCII keyboard, 9-inch CRT and a tape cassette all in one lightweight unit. Fully-programmable in extended BASIC (20% faster than most other BASICs).

14k ROM (Read Only Memory). 8K RAM (Random Access Memory). Expandable to 32K. PET's very own graphic instruction set.

For HOME/OFFICE/GAMES $795 (8K) Portable, Affordable, and Unbelievable.

ONLY $795

PET SOFTWARE NOW AVAILABLE!

MORTGAGE ($15.95) This program calculates mortgage information when provided with certain basic data. Calculated information includes: Principal paid to date; Interest paid to date; Total of Payments paid to date; Outstanding principal and interest; Total payments remaining; Mortgage equity; Interest and principal breakdown for any month. **CHECKBOOK ($15.95)** A cash receipts and disbursements program that will make it a breeze to keep accounts and up to date records. Checks can be searched and sorted by type, e.g., medical, legal, tax-deductible, rent, food, etc. **FINANCE (12.95)** A variety of useful financial formulas in one simple easy to use program. Includes: Compound interest; Discounts; Nominal and effective interest; Annuities; Loans; Depreciation;Earned interest and much more.
ANNUAL REPORT ANALYZER ($22.95) With Annual Report in hand, you input revenue and income figures for previous five years (estimated earnings, too, if you wish) as well as basic Balance Sheet data. This Street Ware program computes: Percentage year-to-year growth in sales, profits, and earnings per share; Average earnings per share and compound earnings per share over 5 years; PE Ratio; Profit margin for previous 5 years, with a graphic display that plots revenues against profit margins; Current ratio; Book value; Return on equity; Debt to equity ratio ; Payout ratio; Dividend yield; Implied growth rate; Implied total return; Theoretical PE ratio; Theoretical value for stock.
STOCK ANALYZER ($34.95) This tape includes a copy of ANNUAL REPORT ANALYZER on reverse side. The program is essentially the same except that data is automatically read from Data Base tapes simply by entering ticker symbols. **DATA BASE (Updated monthly; total of 12 tapes per year) ($175.00)** Includes statistical data on over 2.500 Industrial Stocks on the New York, American, and Over the Counter Exchanges. Data base tapes are updated monthly by stock exchange on a rotating basis, i.e., twelve tapes per year. Data includes: Ticker symbol, Corporate name, Industrial classi-- fication; Revenues, earnings, and earnings per share for previous 5 years; Estimated earnings for current year; Shares outstanding, current assets, current liabilities; Dividends, long-term debt. **OPTIONS ($24.95)** The National Corporate Sciences' version of the Black-Scholes equation, this program computes the theoretical value of an option. The program can be used to equal advantage by both options buyers and options writers. Value of option is graphically depicted by movement in stock price and days to expiration. **BONDS ($9.95)** A variety of bond programs to calculate interest and yield to maturity, present value and future value of bonds, effective yield, and basis price of bonds.

NEW PET PRINTER!

The word is out . . . . The PET INTELLIGENT IMPACT PRINTER WILL HIT THE MARKET SEPT/OCT 1978!!!

SPECS: 60 characters per second . . . . 8½ roll, fanfold, or cut page paper . . . 80 characters per line . . . 7 x 8 dot matrix characters . . . full formatting and forms capability . . . makes up to 4 copies simultaneously . . .
Yes! Prints all PET graphic characters, too!!!

To ensure the fastest delivery possible, just send a $25 reservation deposit. This fully-refundable deposit will reserve you a prime spot in the sure-to-be mad rush for PET PRINTERS.

Only $695

THE COMPUTER FACTORY
485 LEXINGTON AVENUE, BET. 47 & 48 STS.,
NEW YORK, NY 10017
(212) 249-1666 or (212) PET 2001 T-F 10-6 Sat. 10-4

master charge
BANKAMERICARD

CIRCLE 8

Chess Notes

Dear Sir:

In your August issue, you refer in the story on page 87 to the program Sargon which was the winner of the first microcomputer chess tournament, as "a program for the Z-80 developed by a husband and wife programming team." On the following page you have a story that states that Kathe and Don Spracklen have published the literature on their Sargon program, which "runs on a Motorola 6800 microcomputer and won first place at the San Jose Microcomputer Chess Tournament."

I assume the two programs are the same. Is the Z-80 the same chip as the Motorola 6800? Probably not. I think that it is the job of the editor(s) to catch inconsistencies such as this and get them corrected before publication. Otherwise, it is difficult for the reader to have much faith in what he reads in the magazine.

Has anyone made any progress in programming GO (a task that strikes me as incredibly more difficult than chess) or bridge (probably easier)?

Michael W. Ham
Iowa City, IA

Editor's note: The Sargon inconsistencies you noted were due to two separate reports received from the tournament site, both from reliable sources. We concluded that Sargon had been originally *developed* for a Z-80 but, for the tournament, had been rewritten to run on an available Motorola 6800. The two micros, as you point out, are indeed two separate breeds and not interchangeable. The correct information is that Sargon *was* run on a Z-80 at the tournament.

There is only one computer program on GO, to our knowledge, and that is at the University of Wisconsin, Dept. of Computer Science, Madison, WI 53706. Bill Wickert, author of that program, has, so far, received no takers on his challenge for other computerists to come up with a GO program that could match wits with his. As you point out, GO is incredibly more difficult than chess, which explains why there is apparently little computer activity in this field. Although programs have been written for computer bridge, no one has yet advanced this activity into the gaming stage. It could be interesting to watch four computers sitting around a card table playing a few rubbers of bridge. One can only guess at the number of transistors that will blow when one computer trumps his partner's ace!

– H. S.

Printer Info

Dear Editors:

I would like to get more information on the SCI Systems Model 1100 rotary printer which appeared in your "Printers" article in the June issue. However, I cannot locate the company's address. Could you help?

Tim Rand
Storrs, CT

Editor's note: SCI Systems, 8600 South Memorial Parkway, Huntsville, AL 35802. *– G.W D.*

North Star Micro Error

Dear Editor:

I believe an error was made in step 4 of Stephen Pereira's article on the North Star Micro Disk system in the July '78 issue of *PC.*

Step 4 in the article called for returning to the BASIC operating mode from DOS by using JP2A00. Unless his system is different from mine, you would use the command JP2A04 to return to BASIC without destroying (eliminating) the newly debugged program.

Have you experienced the sheer exasperation of spending hours debugging a program and wiping it out with a keystroke?

An operating tip that might help others is to prepare the disk directory ahead of time, with more storage assigned than might be needed, and as you debug the program reenter it as you progress with the debugging. After the program is fully debugged you can then store it on another disk, and compact the disk used while working the program.

Ed Benford
Hawthorne, NY

Author's note: Indeed, I've let a terrible mistake slip through! Step 4 in my drawn-out procedure for creating and saving a BASIC program *should* be JP2A04. As you can tell from the context of the step, I did mean to say that, but as Murphy's law states, those things that you know are absolutely correct – will be wrong.

I hope I've not seriously led anyone astray, and I believe that North Star's documentation would have set any user straight on this point.

– Stephen Pereira

Canada Comments

Dear Sir:

Items move faster through our mail system if you include the Postal Code. The format is letter, number, letter, *space,* number, letter, number. Although this may not be quite as simple as your U.S. ZIP code, it does mean that a letter addressed, Andrew Bates, Canada V6S 1B2, will be delivered to me me. The postal code pinpoints the side of the street in a residential block and even the floor of a building in a business district. How's that for precise!

Software writers take note: We Canadians need at *least* 6 characters for the postal code and 4 characters for the province (state). And if you are going to check the ZIP for all numbers, please put the check in a subroutine so we can replace it with a suitable check for our postal code.

Another small request for software writers who are mailing things to Canada: If your package costs $75, and is distributed on North Star Diskette, for instance, please mark the customs declaration as: DISKETTE $6, PRINTED MATTER $69.

If you mark the price as $75, we end up paying duty on the *diskette* as thought it cost $75. Printed matter comes across the border duty free and there is no duty on an item of less than $10 value.

Andrew Bates
British Columbia, Canada

Blows Against the Timesharing Empire

BY JOHN WALKER

The development of microcomputers has, recapitulated the development of large computers of the Fifties and the minicomputers of the Sixties. The first systems had tiny amounts of memory; were operated at the bit level with switches and lights; and were built and used mostly by people with intimate knowledge of hardware. As the price of memory declined and an infrastructure of software was built, systems advanced to the point where they could be used by software-oriented people, programming in either BASIC or assembly languages. The past year has seen the introduction of complete application packages for microcomputers and turnkey microcomputer systems.

As the declining price of semiconductors continues its asymptotic skid towards zero, the products of ingenuity from peripheral designers seem never to stop. In a couple of years one may be able to assemble a system comparable to contemporary medium-scale mainframes for a price not much more than today's personal computers. Wonderful, but what do we do with it? Obviously, because we're repeating history so faithfully, we build a timesharing system. This is a pretty safe prediction (it being a lot easier to predict the present than the future) because at least three microcomputer manufacturers are offering timesharing systems already, and undoubtedly many more will be thundering to the finish line. This is a terrible mistake.

To understand why timesharing and microcomputers should not be mixed, we must look at the history of timesharing and the reasons behind its development and current wide acceptance in the computer community. From almost inception, computers, it was realized, could be used most effectively in a "hands-on" fast-response environment. Enormous costs of early computers, however, compelled owners to convert all available time to productive use. Rather than have a multi-megabuck computer sit idle while a programmer scratched his head, jobs were assembled in batches and run one after another with no allowances for human interactions.

The expensive components in early computers were the processor (CPU) and memory. It soon became obvious that many programs were being hindered by slow Input/Output devices. It also became obvious that the capacity of the expensive components was being wasted. To alleviate this problem, multiprogramming systems were developed which allowed several programs to run simultaneously in one computer. When one program had to wait for I/O to complete, the processor could be used to handle another program. Multiprogramming required more memory than purely serial execution, but because it used tremendously expensive CPU more heavily and did more work in a system of slightly higher cost, it paid for itself.

One programmer put his fist through a keyboard after the computer didn't respond for several minutes.

Multiprogramming, which originally shared the CPU among programs sharing memory, could be extended to share memory among programs which could not all fit into memory simultaneously. With fast disk of drum storage available, a program encountering a substantial delay could be written out and another program brought in to use the expensive memory and CPU. This swapping of programs allowed delays longer than a simple I/O wait to be accomodated without tying up expensive memory.

With continued refinement of multiprogramming, a solution was found for providing "hands-on" computing without dedicating an horrendously expensive computer to each user. If enough users could be collected together and connected to one computer, whenever one user was thinking, typing-in, or receiving a response from the computer, that machine could process work submitted by another user and thus keep the CPU busy. Because CPU speed is so great and almost no processor time is needed to type out information on the user terminal or receive characters from a keyboard, each user generates a minimal amount of time over a long period. The timesharing user tends to request short bursts of computing separated by waiting-times measured in seconds. Hence, in theory, the user will not notice that he is sharing a computer with many people. He will have the illusion that he is the master of a dedicated computer, ready to respond to his demands. Finally, by inducing each user of a timesharing system to pay a relatively low price user fee, say $20/hour, the high cost of the system can be recovered.

We see, then, that the driving force that created timesharing was economic; finding a way to provide many users with interactive computing even though computers and memory were very costly. Once established, a timesharing utility can provide many useful services. Since many users share a common system, they also can share programs and data files.

Now the same benefits are being offered microcomputer users with the advent of microcomputer timesharing systems. However, there is no need to go to timesharing to reap its benefits. In effect, we must change our attitude about computers; we must learn to view communication, not processing, as the key. Timesharing, as already defined, is the sharing of central processors and memory among a number of users so that that all users do not simultaneously receive service. With the price of processors and memory dropping, the practice that made sense in the Sixties is stupid in the Seventies. Timesharing is widely identified with benefits that evolve from its adoption on large computers. However, because the U.S. proceeded from stagecoaches to trains to airplanes for long distance travel doesn't mean a developing country today has to go

through the same steps. Microcomputer users should pause and think before following the path big system people took into timesharing.

What's so bad about timesharing, anyway? Let's take a look at the technical problems that plague timesharing. The most serious problem stems from the fact that timesharing is, after all, an illusion. As long as people "play by the rules" and interact on a line-by-line basis with little computation, everything works fine. Before long, somebody will ask for a substantial amount of computation, such as asking the computer to determine whether 2,363,383,927,257 is prime. Because you expect this procedure to take a while, you're not upset when the answer doesn't come back right away, and because your computation can "mop up" time not being used by other people, all is well. Suppose, however, that somebody else has a burning desire to know the 987th digit of pi right at the same time you started your program. Now we have two mops contending for one CPU, and somebody is going to lose. If the time is equally split, your program will now take twice as long to run. And what about you? You're sitting there fuming at the terminal wondering why what took a minute yesterday takes two (or five or ten or twenty) today. The concept of timesharing finds its own negation in the following slightly cynical "Laws of Timesharing" (with apologies to C. Northcote Parkinson):

1. Computations grow in complexity to fill existing processor power.

2. Programs grow in size to fill existing memory.

If these two widely-observed rules hold, the self-aggrandizement of programs in a timesharing system will eventually lead to inter-program contention that degrades response time. As a result:

3. Timesharing is characterized by unpredictable variation in response time.

The unpredictability of response time is the most frustrating aspect of timesharing. Large systems can be beautiful for obtaining a fast answer to a big problem. But the system can be singularly infuriating when it pauses for up to a minute when asked to do something like "list my five line program" that a microcomputer will always do instantly. (Long unpredictable delays make some people positively homicidal. I know one programmer who put his fist through a terminal keyboard after waiting several minutes for a system to respond.)

Another problem in timesharing is the catastrophic impact of a system crash or breach of security on its users. As systems grow more and more complex, there are more hardware components to break and more dark corners of software in which bugs may lurk. As a result, large timesharing systems tend to be less reliable than small dedicated systems. Besides, they are hellishly difficult to maintain and enhance. When the system fails, many users are simultaneously infuriated. If the users are all in the same location, mobs dangerous to the health and welfare of systems programmers can form. When many users share a system, they may also share programs and data. Unfortunately, unless the system is very careful, they may also steal or destroy each other's programs or data. Millions of dollars of Government money have been used to develop systems that do not have these problems. But no system exists that allows both cooperation and privacy. Some very good systems have been developed, but none of them has remained secure against pilferage attempts of clever programmers.

Microcomputers now in use make timesharing obsolete...

The microcomputers we now use have made timesharing obsolete. It is absolutely ludicrous to have four users sharing a computer that costs less than the terminal each one is using to access the system. We should take advantage of our position in the "third invention" of computers and leapfrog timesharing to achieve its lofty goals. If we develop networks to allow microcomputers to intercommunicate, we can then share programs and data without encountering the infuriating problems of timesharing. We must view communication as the dominant aspect of a computer's function, and build software and hardware structures based on this concept. Some systems embodying this concept have already been built and may serve as a prototype for future developments.

By proper application of networking, we can build systems that provide computers to perform most tasks rapidly, reliably and predictably. It should be a system built to the scale of a human being, not a corporation. Furthermore, it should be a system that a person can understand, can use effectively, and can maintain without managers, supervisors, inter-office memoranda, PERT charts, or a horde of narrow specialists. Such a system will be a network of intelligent components, and will intercommunicate with each other to allow the sharing and communication provided by current central timesharing systems. Because other users will be "at arm's length" over a network, it will be easier to build systems that are probably secure compared to complex centralized systems where interfaces between programs are numerous and complex. Most user tasks can be done locally, on the user's own machine. Only when communication is required need a request be made that could cause unwanted delay.

Changing economics of computer hardware now have solutions to problems that timesharing never solved. Small system people like ourselves have the outlook, experience and independence to build our own systems. These special designs will return computers to the people they were built to help in the first place. It has been said that computer software systems tend to reflect the structure of the organization that produces them. Is it little wonder that most existing systems are centralized and hierachical. Too many systems that promise "distributed processing" deliver, instead, a more complex terminal connected to an even bigger central site. Microcomputing, with its thousands of basement inventors and garage-shop entrepreneurs, has the "organizational structure" to build loose, distributed, egalitarian, flexible and powerful systems. Let's do it. □

John Walker has been involved in the design, implementation, optimization and support of large-scale timesharing systems for over eight years. Recently, as a partner in Marinchip Systems, he has worked on developing a distributed network operating system based on the ideas in this article.

RANDOM ACCESS

Computer Reveals Galaxies Shaped Like Candy Bars

Using one of the world's most powerful computers as a miniature universe, man is now "seeing" the birth and evolution of stars and galaxies, including his own.

Two-hundred million years are compressed into several hours computer time — and the distance light travels in 100,000 years is reduced to the length of a television screen — to witness galaxies collide and gas clouds beget embryonic stars.

These simulated cosmic events threaten to alter traditional ideas of how galaxies form, what shapes they assume and what happens when they collide.

Elliptical galaxies were thought to be oblate — shaped like a frisbee. But computer-generated elliptical galaxies at NASA's Ames Research Center in Mountain View, CA, reveal their three-dimensional shape to be prolate — oblong, like a candy bar. And astronomers are lately reporting observational evidence confirming the computer's discovery.

Ames' researchers believe that computer simulation of cosmic events is an important factor in fully understanding the dynamics of the universe. Astronomers now have a "laboratory" to test theories and conduct experiments.

A scientific team at Ames led by University of Chicago astronomer Dr. Richard Miller and Ames astrophysicist Dr. Bruce Smith are instructing Ames' Illiac IV computer to create swirling systems of 100,000 computer points. Each point represents the mass of about one million of our Suns to give the simulated galaxy a realistic total mass. Each point "feels" the gravitational tugs of its neighbors and "absorbs" energy from exploding supernova.

Miller and Smith routinely follow up to 120,000 "stars" which are free to move about in a "cube" — a three dimensional boundary of space created by the computer corresponding to 100,000 light years from edge to edge. They can order Illiac IV to rotate the cube so they can watch events from every angle. They can view swirling dust clouds from below or peer down at gigantic galactic collisions.

Scientists at Ames have filmed several computer simulations.

The Ames team suggests that elliptical galaxies seen through the telescope are really the same prolate bars generated by the computer but seen in various projections. Astronomers view snapshots of galaxies projected onto the sky. They have only one view of the galaxy and can only guess its shape, since it would take millions of years to track the galaxy across the sky and view it from a different perspective.

But with a powerful computer, they can "examine" such galaxies from every direction and easily visualize their three-dimensional forms. Confirmation of the computer's revelations concerning the shape of elliptical galaxies has now been reported, based on observed velocities of the galaxy's stars.

When Miller and Smith set two 50,000-star galaxies on a collision course, there is much greater interaction between the colliding galaxies than was forcast. The galaxies first contract, their gravitational fields reinforcing each other. But then, they bounce back in a violent expansion that flings hundreds of stars out of the galaxies. The end result is a merger of the two galaxies but with considerably fewer stars.

Curiously, spiral-shaped galaxies such as the one to which Earth belongs never appear for long without assuming the prolate bar shape. This perplexing situation indicates present knowledge concerning the physics of spiral galaxies is inadequate. Other basic physical processes must play a role in the formation of stable spiral galaxies.

Ames researchers are now following simulated masses of gas as they coalesce and form stars. Some stars are programmed to

have a short life, collapsing into brilliant supernova eruptions, spewing their masses out into the primordial galaxy. In this area, too, Miller and Smith expect surprises from Illiac IV. Perhaps more theories concerning the nature and evolution of the universe may change as the gigantic computer reconstructs events which took place at the beginning of time.

Prisoner's Dilemma

If you're looking for a challenge for your strategic instincts and programming skills, the tournament for the Prisoner's Dilemma may be for you.

This national computer tournament is based on a nifty little game called the Prisoner's Dilemma. There are two players, but unlike most games, the players are not in total conflict. In fact, both can do well or both can do poorly.

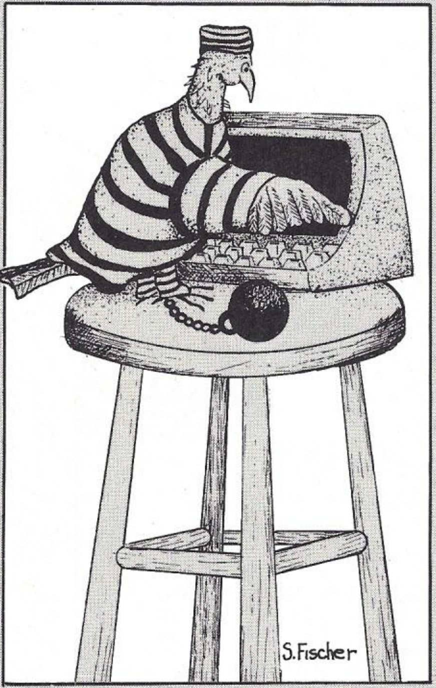
The game will be played for an average of 200 moves, and in each move, each player can choose either to cooperate or to defect. If both cooperate, both do well. But if one defects while the other cooperates, the defecting player gets his highest payoff, and the cooperating player gets taken for a sucker and gets his lowest payoff. The catch is that if both defect, both do poorly.

Payoffs in the tournament for a given move are 3 points each if both cooperate; 5 points to a player who defects while the other cooperates, with 0 points to the sucker; and 1 point each if they both defect. The score of a player in a single game is his or her total over all moves.

To win the tournament you have to get the highest total score summed over all the games you play. Therefore your object is to get a good score in each separate game, but not necessarily to get a better score than the player with whom you are currently playing.

To join the computer tournament, submit a program written in BASIC or FORTRAN IV which will be a decision rule — a strategy — for selecting the "cooperate" or "defect" choice at each move. The decision rule may be based on the history of the game so far. For example, a simple and effective decision rule is "Tit for Tat": cooperate on the first move, and then do exactly what the other player did on the previous move. Quite sophisticated decision rules can be written in as little as 25 lines.

This tournament for which there is no entrance fee, is part of a research project to understand the nature of skillful performance in a two-sided environment which is partially cooperative and partially competitive.



Each person who completes an entry will receive a report describing the results of the tournament. The winner will receive a report describing the results of the tournament. The winner will receive an engraved trophy.

For more details write to Professor Robert Axelrod, Institute of Public Policy Studies, The University of Michigan, 506 E. Liberty St., Ann Arbor,MI 48104.

Literature Update

Several publications of interest to personal computer users came to our attention recently.

The Recreational Programmer serves users of both personal computers and programmable calculators. The bimonthly magazine is not aimed at any particular machine or logic system. Editors welcome reader submissions in AOS, RPN, Tiny BASIC, PILOT and FORTRAN, and pay for programs published. Subscriptions cost $12 per year. Write to *The Recreational Programmer,* Box 2571, Kalamazoo, MI, 49003.

Polyphony, an applications-oriented publication for experimental electronic musicians, covers all aspects of electronic music from music theory and notation to microprocessor-controlled polyphonic synthesizers and software. Published quarterly, *Polyphony* includes construction projects, original circuit design, modifications or expansion of commercial equipment, book/record/product reviews and synthesizer patch charts. Subscriptions cost $4 per year in the U.S.; $5 in Canada and Mexico; and $6 in other countries. For a free sample copy, write *Polyphony,* P.O. Box 20305, Oklahoma City, OK 73156.

The Imsaider, a customer newsletter from Imsai, is now issued bi-monthly in a glossy magazine format. The magazine, which publishes information on Imsai products, is available to Imsai customers for $4 per year. Some dealers offer the *Imsaider* for $1 per copy. The magazine welcomes articles and letters from Imsai users concerning applications, software enhancements, experience with hardware, user groups, and seminars and events of general interest. Contact Imsai Manufacturing Corp., 14860 Wicks Blvd., San Leandro, CA 94577; (415) 483-2093.

Quest Electronics publishes *Questdata,* a monthly newsletter supporting their COSMAC Super

Elf and other RCA 1802 based systems. Typical articles include machine language experiments, video graphics software and Tiny BASIC. Order your $12 subscription from *Questdata,* P.O. Box 4430, Santa Clara, CA 95054.

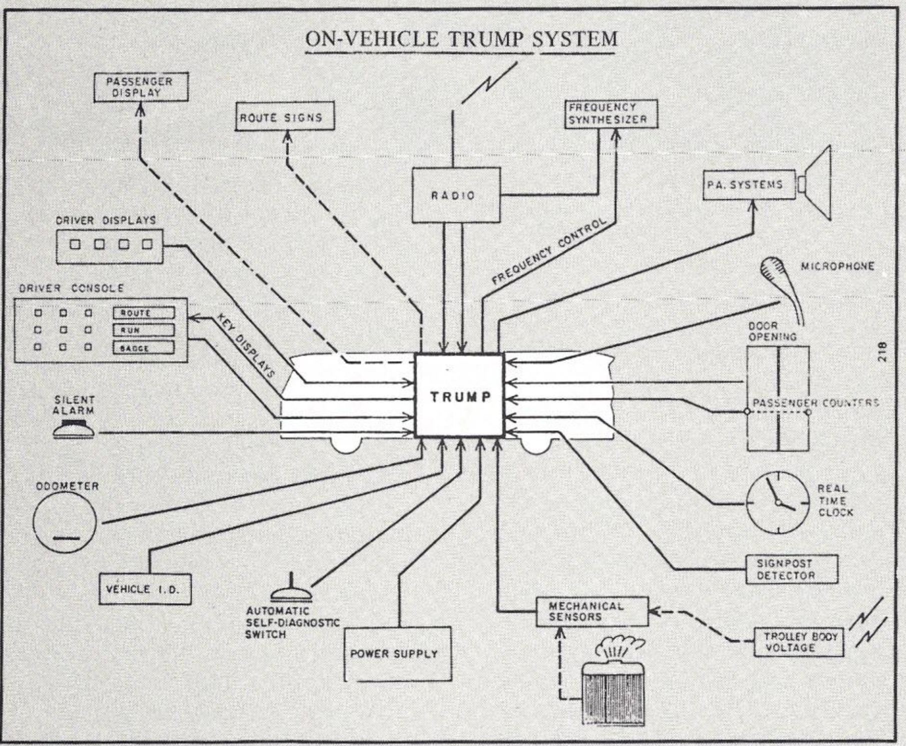
Radio Shack computer owners may want to order Richard Huebner's *Guide to TRS-80 Information.* The 20-page *Guide* includes an index to magazine articles about the TRS-80 and a list of hardware and software sources. One *Guide* costs $3; 2 to 9 cost $2.50 each; and 10 or more cost $1.90 each. Send your check or money order to F. E. Huebner, P. O. Box 37206, Oak Park, MI 48237.

Micros Run Into A Traffic Jam

At the Canadian Information Processing Society's (CIPS) Annual meeting in Edmonton, R.B. Rebeiro, of the IBI Group of Edmonton, Alberta, reported the result of a survey of public transport problems. He concluded that microprocessors would eventually solve the mushrooming traffic problem in expanding metropolitan areas. Twenty-two municipalities across Canada were scrutinized for existing problems; the data procured formed the basis for this report. Problems were found to exist in control of taxi, police, fire, ambulance, transit and traffic services. Mr. Rebeiro cited some typical examples and remedies where his IBI had used microprocessors to alleviate a worsening situation.

"Urban system managements," observed Mr. Rebeiro, "are being asked to do more with less." Examples he cited included the following:

1) Priority Control. Congestion in urban areas hinders emergency service. It is necessary to provide emergency vehicles with priority lanes at critical areas on roads to maintain emergency-response effectiveness. One solution is the installation of priority-generating microprocessors in appropriate vehicles to instantly create, when necessary, priority lanes in a computerized traffic system.



2) Variable Message Signing. The process of controlling bumper-to-bumper automobile traffic by microprocessors would allow for lane control at peak periods, prohibiting entry into certain parts of traffic networks at certain times of day, etc. This could be accomplished by inserting micros in all automobiles as well as in the traffic system itself. Tokyo, currently is experimenting with such a system to help solve its traffic problems, reportedly among the worst in the world. On-road conditions are displayed by micros in cars travelling through Tokyo. These in-car displays inform drivers of potential road problems, routing information and congestion. Detroit's growing enchantment with microprocessor applications may make such a travel companion and traffic navigator a reality within the next ten years.

3) Automated Fare Collection. Increased use of passes and credit cards in transit systems make the microprocessor an ideal fare collector in all types of public transportation.

4) Automated Alert. A coaxial cable, linking every dwelling unit in a community to a central emergency control, is a promising use of micros. Linked into central control would be fire and burglar alarms as well as medical emergency services when needed. Such a system is actually in test operation in at least one North American community.

5) Portable Data Recorders. Presently traffic engineers must rely on traffic-flow measurements through pneumatic tubes laid across roadways. This system says Rebeiro, is unreliable. A microcomputer system with a cassettle or bubble memory, and with improved sensor devices would provide a more effective mechanism, he says.

6) Traffic Monitoring and Area Control. Traffic engineers are going to have to be convinced that controlling traffic flow does not require the use of an IBM 370 or a PDP 11. Relatively sophisticated local controls can be coordinated by microcompouters to achieve better sequencing of all traffic signals.

Storing Data With A Laser

Lasers may one day be used to store data in computers, according to scientists at IBM's San Jose Research Laboratory.

The scientists — George Castro, Dietrich Haarer, Roger Macfarlane and Peter Trommsdorf — were recently awarded a patent for a method of using a tunable dye laser to store information. Based on a photochemical process called "hole burning" the technique increases the amount of information that can be packed into a given space.

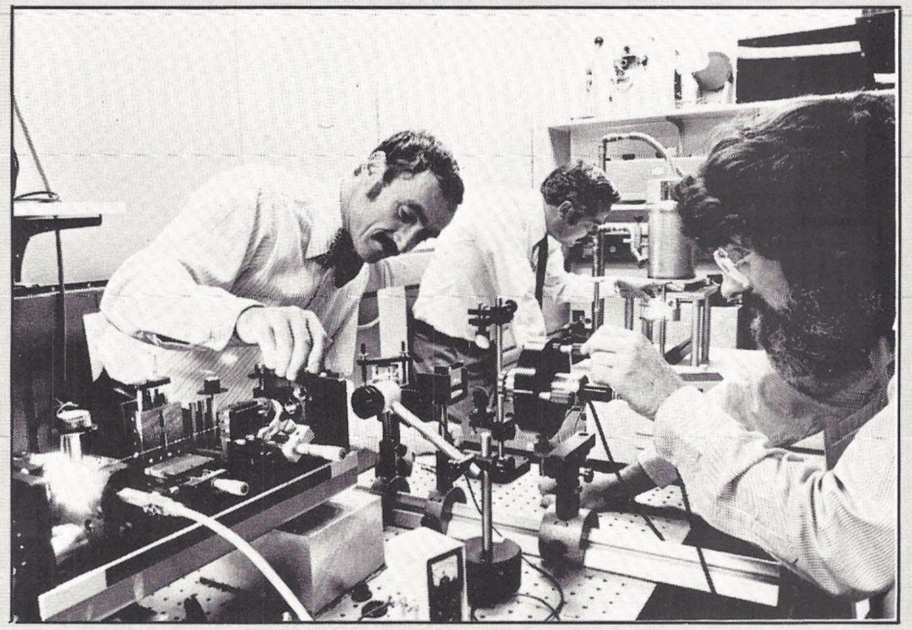
In the system described by the patent, each "bit" of computer data is identified by its location in the frequency spectrum as well as by its location in two-or three-dimensional space. Potentially, many thousands of the "frequency coded" bits could be stored at a single, microscopically small region in space. The size of these regions could be as small as the "diffraction limit" of a laser beam, a dimension in the order of one micrometer (1/25,000 of an inch).

The invention uses a laser that can be tuned — like adjusting the station selector on a radio — so it emits light of various colors, but whose light rays at a particular setting of the dial all oscillate within an extremely narrow frequency band. The tunable dye laser was itself invented in 1966 by IBM researchers.

Highly monochromatic light from such a laser shines on a sample of a photoreactive material cooled to a few degrees above absolute zero. This action produces a chemical change in a small percentage of the material's molecules — just those molecules which can absorb energy at the frequency of the laser light.

As a result of the chemical reaction, the frequency-selected molecules are changed into a new compound and there are no longer any of the original molecules left to absorb light at the frequency of the laser light.

As a result of the chemical reaction, the frequency-selected molecules are changed into a new compound and there are no longer any of the original molecules left to absorb light at the frequency that produced the reaction. The phenomenon is observed as the forming of a gap, or "hole", in one of the material's optical absorption peaks —hence the name "hole burning".



These absorption peaks can be thought of as features in a graphical "fingerprint" that distinguishes one material from another. They occur when light waves cause certain atoms to absorb part of the radiation and vibrate in concert with the light waves. The effect is conceptually similar to the resonant vibrations formed in a tuning fork in response to sound waves of a certain frequency.

In experiments performed to test the feasibility of the hole-burning technique for data storage, the scientists were able to burn a hole whose width (in frequency units) was less than one thousandth the width of the absorption peak in which it appeared — one of the narrowest optical features ever observed in a solid.

By implication, the peak is wide enough with respect to the laser frequency to allow burning one thousand holes side by side. This burning could be done sequentially simply by tuning the laser beam to one thousand different colors, or frequencies. The "ones" and "zeros" by which information is encoded in a computer could then be represented by the presence or absence of holes in the absorption peak at particular frequencies. The encoded information could subsequently be read out by reducing the power of the laser and varying its frequency over the same range used for writing.

The photoreactive molecules being investigated at IBM are suspended in tubes of frozen organic liquids or polymers at very low temperatures and concentrations (about 10 photoreactive "guest" molecules per million molecules of the "host"). Within the organic matrix, each photoreactive molecule assumes a random orientation; this randomness produces an "inhomogeneous" bell-shaped absorption peak like the familiar normal curve used in statistics. This peak actually consists of many narrow, overlapping spikes known as "homogeneous" absorptions, each of which corresponds to the resonance from photoreactive molecules having a particular spatial orientation in the organic liquid matrix.

To make room for many holes

across the width of an inhomogeneous peak, it's necessary to keep the width of the homogeneous peaks as narrow as possible. Since this width is directly related to the thermal agitation of the molecules, the material must be cooled to nearly absolute zero — the temperature at which thermal motion ceases.

According to the patent, one possible implementation of a hole-burning storage system would be an array of tiny blocks, or elements, of storage material deposited on a supporting structure. In such a hole-burning system, assuming the elements are closely packed and each is one micrometer in diameter, the density of elements would be 100 million per square centimeter. With 1000 bits stored in each element, the contents of the human brain — which has been estimated at a million billion bits — would be able to fit into a storage space of about one square meter.

Work on the new storage technique is still at a very early stage of research. "We have a long way to go before this technique can be made into a technology," said Dr. George Castro, manager of the San Jose laboratory's physical science department. "So far we are investigating a variety of potential storage materials and trying to understand the temperature limitations of the phenomena. We still have to address the many unsolved problems that have prevented other optical methods of storage from becoming technologies. In fact, optical methods of storing information were expected to lose an important advantage, namely that of high storage density, as the storage densities in the existing magnetic recording technologies approach optical limits in the next decade. Hole burning breathes new life into optical storage by extending that density a few orders of magnitude."

Bytes And Stings

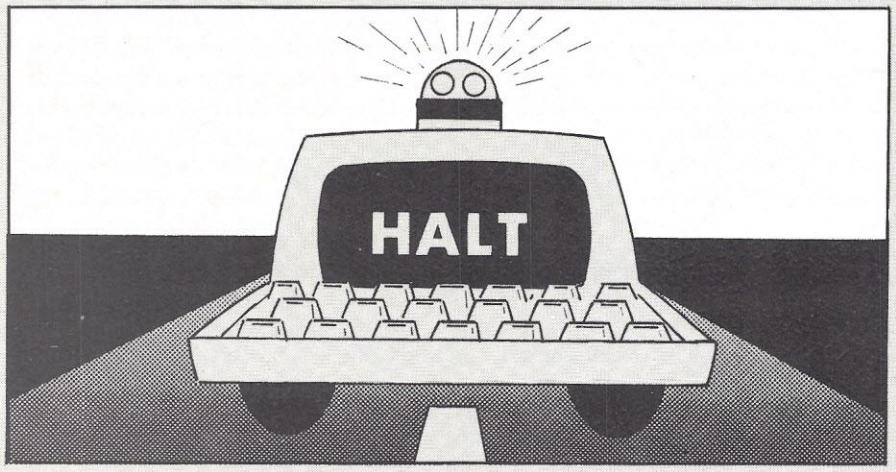
Ryal C. Poppa, president of Pertec Computer Corporation, thinks bank computers can stop some con-artists from putting "the sting" on their victims.

Known by such intriguing names as "pigeon drop", "Jamaican trust" and "charity switch", the swindlers perpetrated by these con-artists share some common elements that provide the key for bank and savings and loan personnel. Generally, the victim is told a plausible story by a stranger and led to believe that some fast money can be made. In each case, the victim is convinced to withdraw money from the financial institution, and in many instances the citizen withdraws all of his or her life savings.

A switch is then made. The victim thinks he's holding his own money as well as the con artist's, but eventually discovers he's holding only cut paper. The swindler is long gone with the real money.

"Most computers at financial institutions easily could be programmed to give tellers immediate access to information needed to spot potential victims of fraud schemes: data on the size and frequency of transactions in each account," Poppa noted. "In fact, the computer even could be programmed to alert the teller automatically whenever an unusual transaction was requested.

"For example, Peter Pitchess, Sheriff of Los Angeles County, recently stated that when most or all of the funds are being removed from a relatively inactive account, the teller should refer the customer to an officer, such as the branch manager, before handing over the requested funds." If the officer suspects a fraud game, the local law enforcement agency could be called immediately, Poppa said.



Before the advent of the computer, information on the activity level of an account was difficult to make readily available to tellers, the PCC executive commented.

"Now, however, firms in the computer industry also have the skills and tools to tailor programs to deal with the quality of human existence in addition to meeting business and technolgical needs that consumed most of their developmental resources in the past," Poppa added. "The data processing industry has the capabilities to assist in the prevention of bilking schems that have been around for decades, and industry leaders are becoming aware of the positive force that can be employed," he commented.

"We often are told that law enforcement is everyone's business — that crime can't be reduced unless citizens take an active role. The computer is now providing financial personnel with a unique opportunity to do just that: cut down consumer fraud," Poppa concluded.

Club News

An Apple II Users Club was recently formed in Portland, OR. Cleverly acronymed APPLE (Apple Portland Program Library Exchange), the group seeks to exchange programs and ideas with other clubs and individuals. For more information, contact Ken Hoggatt, President, 9195 S. W. Elrose Ct., Tigard, OR 97223; (503) 639-5505 (home); (503) 644-0161 (work).

A PET Users Group has been formed in Vancouver, Canada, with current membership at 40. Club format includes a short presentation by an owner on programming or hardware, news from Commodore and program swapping. If you're interested, call Rick Leon at (604) 734-2060 (home), (604) 324-0505 (work); or write: Vancouver PET Users Group, Box 35353 Station E, Vancouver, B.C., Canada.

Organized by readers of the *Calculator Lib* newsletter, the Liberated Calculator-User's Club, an independent non-profit group, was formed to share members' knowledge of calculators and related fields, to create a forum for members to meet and identify with each other's interests, and to disseminate ideas about calculator arithmetic.

Present members communicate with the club in English, French German and Hungarian. The club hopes to expand its rolls to include members in Belgium, Canada, Denmark, India, Japan, Singapore and Switzerland.

The club seeks volunteers to act as officers in the editorial committee, correspondents, reporters and translators. Location and experience are not important, and volunteers will set their own working hours; no salaries will be paid.

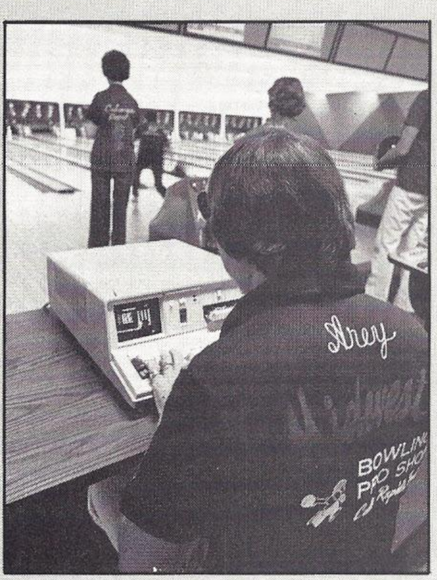
For more information, contact Gene Hegedus, P.O. Box 2151, Oxnard, CA 93034; (805) 486-7191.

A release of inexpensive software was initiated by the North Star Users Group recently. Over 300 programs on 20 diskettes are now available for a small copying charge plus the price of a new diskette. The disks run in microcomputer systems utilizing North Star DOS and North Star BASIC. The library includes business and finance programs, mailing list programs, math software programming utilities, a PILOT interpreter and games.

For more information, send an sase to J. Dvorak, Authorized NSUG Program Library Distributor, 704 Solano Ave., Albany, CA 94706.

Bowling With Bytes

A computer is scoring well with bowlers in Cedar Rapids, Iowa. And especially with center owner Steve Benz.



According to Benz, league competition for individuals and teams requires a lot of record keeping for the league secretaries — scores, wins and losses, averages, high games and high series and league standings. What used to take hours is now completed by an IBM 5100 in about fifteen minutes. The computer displays team and individual bowler names in sequence on the screen, and asks for the game scores. It can handle irregularities like make-up games, substitute bowlers and handicap changes.

After the scores have been entered, the 5100 displays a full record of league name, team name bowlers, game scores and total pins. A secretary can get a printout of that day's record or can ask for a printout of the year's cumulative statistics to date.

Israel Opportunities

In a study prepared by IBM and the Chase Manhatten Bank, Israel was ranked first among developing nations in sophistication of computer usage. This ranking was based on a variety of factors, including the number of computers in relation to the society, condition of the local computer manufacturing industry, sophistication of applications and level of training facilities. More than half the firms using computers are concentrated in Tel Aviv, with Jerusalem and Haifa (where data processing is growing faster than anywhere else in the country) the next largest users. There are also opportunities in Beersheba. Of the persons working in the computer field, most are employed as keypunch or computer operators, programmers, systems analysts and managers, though of course there are opportunities in areas like computer librarianship, technical services and scientific research.

A new area is computers in Moshavim. Elazar, in Gush Etzion south of the capital, and Neve Ilan, to the west of Jerusalem, are two moshavim shitufi'im (cooperative settlements) involved in the computer business. The former is the country's first completely industrialized moshav. Among its industries is a computer service bureau, based on an Elbit CR17, an Israeli manufactured minicomputer, which provides data entry and data processing services for clients. Neve Ilan, in addition to its agricultural products, is also involved in software services. (From the "Information Bulletin — Conditions of Employment in Israel for Computer Professionals", published by the Jewish Agency.)

Black-box war

. . . A minor conflict between the country's two most popular computerized-chess-boxes has broken out. Seems that *Chess Challenger-10* played a game against *Boris* at its home-office for the announced intention of testing its new, updated, *Challenger-10* model against *Boris.* Fidelity Electronics, manufacturers of *Chess Challenger,* claimes it played 7 games against *Boris* and beat *Boris* 7 times. It sent around a copy of only Game No. 5 in which *Boris* looked rather silly. According to a representative of the Chafitz Company, manufacturers of *Boris,* such a game was not possible because it could not be duplicated on another *Boris.* All *Boris',* claimed the representative, have precisely the same program which have (at date of letter) never been changed. The Chafitz head office had a further angry reaction after these initial remarks and submitted the letter below. When Fidelity Electronics saw the Chafitz reply, it responded with its own retort. Both letters are reprinted here. Regardless of what position readers may take after studying the exchange of letters, one fact remains undisputed: both *Chess Challenger* and *Boris* are doing more to popularize the game of chess than even nationally publicized chess matches by international grandmasters for purses in six figures. Both companies deserve to be awarded commendations from the chess world for promoting the game which is currently being played by more than 20 million Americans (estimation).

Letter from Steven R. Chafitz, Pres. Chafitz Co., 1055 First Street, Rockville, MD 20850:

Since the game of chess has always been looked upon with some "reverence", I felt it was important that I send this letter to you.

As you are well aware, there are a variety of personal chess computers available in today's marketplace. During the recent Consumer Electronics Show held this past June at McCormick Place in Chicago, manufacturers of these computer chess games exhibited their product lines.

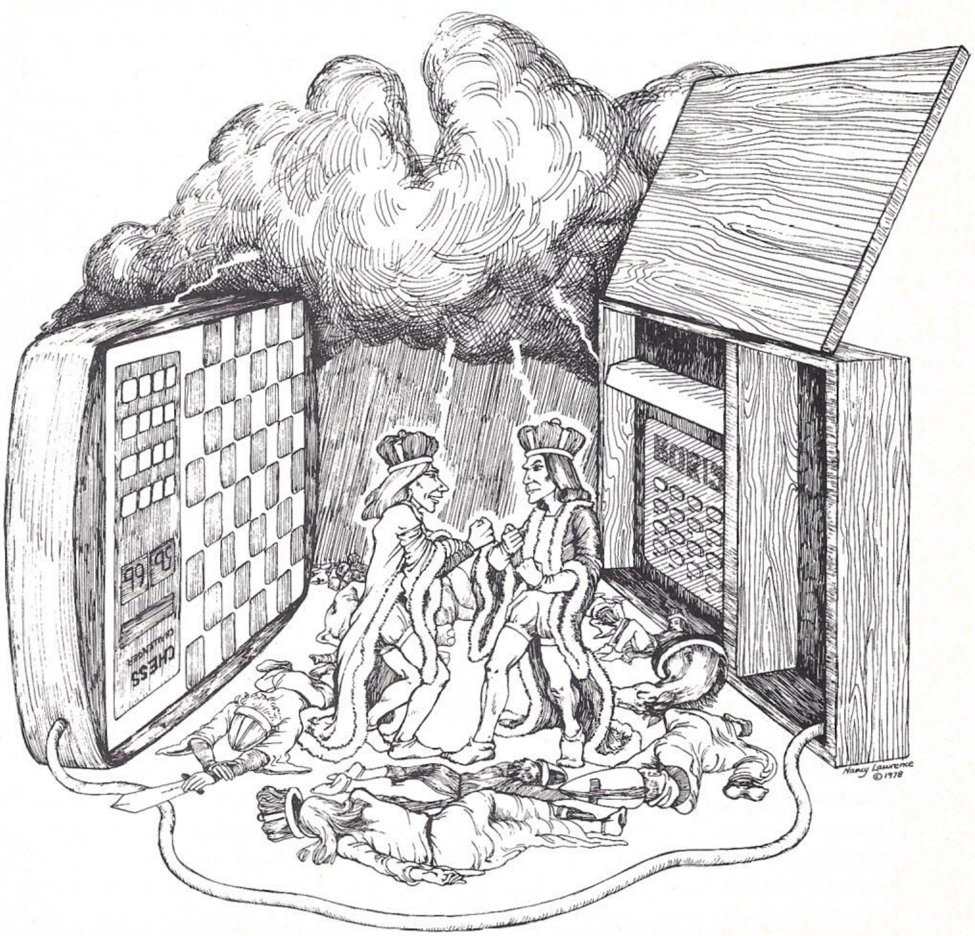
Unfortunately, one of these manufacturers began a propaganda attack on another. The material, which was released at the show, alleged that one chess computer easily defeated another. In evaluating the game spplied by Fidelity Electronics, which claims that their unit defeated BORIS, *we at Chafitz, along with several other recognized chess experts, concluded that the game which they released at the show could not have occurred. Rather than spending much time to discuss this matter in detail, we elected to forward you our analysis of this game and invite you to perform your own "comparative testing".*

It is with reservation that we have decided to disseminate this information to the chess community; however, we feel it is important that the facts be known. Our primary objective is to discourage further practices of this nature, which, in our opinion, would only degrade the high character of chess.

Below is the report which Fidelity claims is a re-cap of a game between BORIS *and Chess Challenger "10".*

THIS GAME WAS NEVER PLAYED! *I invite you to prove for yourself that* BORIS *is incapable of playing this game.*

The pivotal move is #8. If BORIS *made the move, F8-B4, (see Fig.) which Fidelity claims, it leads to the eventual loss of* BORIS' *queen. But* BORIS *will not make this move with his timer set on 3 minutes. Set up the board position shown, on your own* BORIS. *Set his timer at 3 minutes and turn him loose. You'll note that* BORIS *considers F8-B4 but then decides on A5-D2 long before his 3*



minutes are up. This move, of course, leads to a trading of queens. But on his next move, BORIS *will capture the white bishop (F5-E4), thus putting* BORIS *ahead in both position and material.*

At 3 minutes BORIS *will always move A5-D2 because he sees it as clearly the best move. There are six other moves on this sheet which* BORIS *cannot make at 3 minutes. I don't claim that* BORIS *never makes "dumb moves". He may, but he is incapable of making the particular move by which Fidelity claims he lost this game, at 3 minutes.*

We at CHAFITZ *do not feel that game strength is the only important feature in a chess computer.* BORIS *has so many advanced features that he would still be "king" even if a competitor comes up with a unit that could defeat* BORIS. *We also realize that other manufacturers may feel that the only way they can compete with* BORIS *is to claim a stronger game. But we are truly saddened to think that anyone would find it necessary to distribute false information.*

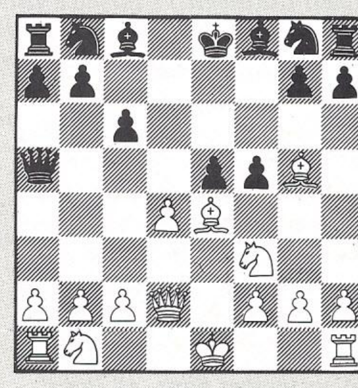
White – *CHESS CHALLENGER 10* Black – *BORIS*

Time set at 3 minutes per move.

1.	e2-e4	e7-e5 *
2.	g1-f3	d7-d5 *
3.	d2-d4	f7-f6 *
4.	f1-b5+	c7-c6
5.	b5-d3	d5-e4
6.	d3-e4	f6-f5
7.	c1-g5	d8-a5
8.	d1-d2	f8-b4 ***
9.	c2-c3	f5-e4 *
10.	c3-b4	a5-d5
11.	b1-c3	e4-f3
12.	c3-d5	f3-g2 *
13.	h1-g1	h7-h6
14.	d5-c7+	e8-d7
15.	d4-e5+	Resign

Comments by Chafitz: "The mark (*) indicates moves that *BORIS* cannot make! and the mark (***) indicates a pivotal move that *BORIS* will **never** make!"

Controversial moment in game between *Boris* and Chess Challenger-10. Position after White's 8th move of D1-D2.



Response from Sidney Samole, Pres. Fidelity Electronics, Ltd. 5245 Diversey Ave., Chicago, IL 60639;

In response to Mr. Chafitz's letter, Fidelity held a tournament between Boris and Chess Challenger "10" only after Boris advertised and displayed comments about Chess Challenger.

As you know, Boris is a computerized chess game which uses PROM's *for its program and, therefore, can be reprogrammed with each unit. Chess Challenger "10" uses* ROM's *and, therefore, all purchasers receive the same quality chess game.*

The Boris unit used in this tournament was purchased at retail from Horchow. We accept the fact that one Boris game may be different from another. However, the game we used did in fact lose seven games out of seven to Chess Challenger "10" and we invite you or any member of your staff at our expense to come to Chicago and examine the unit and replay the games.

Our company, which is a world-renowned manufacturer of sophisticated biomedical equipment supplied to the Veterans Administration and other institutions of this stature, has a long standing reputation that we would never jeapordize by releasing false or misleading information.

Chess chatter

. . . Although not in the field of computer chess, there is another form of chess currently popular among universities that depends on electronic assistance. That activity is telephone chess and the current champions of the National Phone Chess League are the Berkeley Riots of the University of California. There are rumbles challenging a big computer to a telephone match. If any IBM, CDC, etc. would like to take up this challenger against one of these teams in a telephone match, a note to this column will get the lines humming . . . A letter from Roger J. Cappallo, RFD 2 Box 327, Lincoln, MA 01773: "I'm glad to see that there is a coordination of the efforts and thought of the not inconsiderable mass of amateur computer chess enthusiasts. I'm working on my PhD at MIT, analyzing lunar laser ranging data, but have been interested in computer chess for many years. I'm a class B player and a former systems programmer, but have not had the opportunity to write a program that does more than make legal moves (it was written in FORTRAN and run in batch – a pain to work on and started using too much time.) However, I recently bought an Horizon I. As soon as I get an assembler, I plan to write a serious program. Incidentally I have played the Greenblatt program and Tech II (both of which can be crunched positionally) many times. On one occasion I got to play Baylor's program using CHEOPS, the hardware box that generates the move tree. Baylor's program was overly thorough – searching 10 and 11 ply deep in a minor-piece endgame!! I managed to get a draw." . . . Richard Danner, 5045 Galileo Drive, Sierra Vista, AZ 85635 says: "I am interested in microcomputer chess. I have a Poly 88 micro with two chess programs: one in machine code (*Microchess*) and one in BASIC (*Tenberg*). I would like to get more challenging programs. Any recommendations?" . . . "I have a relatively unsophisticated chess program (based on Jenning's *Microchess*)" writes Charles A. Michalski of 820 Wolfram Unit D, Chicago, IL 60657. "It is running on one of the IBM 370 models and holds its own against novice players. But I am currently developing my own original program which hopefully will evolve into a system suitable for national competition." . . . "Has anyone thought about writing a chess program such that the opening and middle game results in all pieces being exchanged and then write a strong endgame program since we

are dealing with fewer types of pieces?" That is the query posed by G.Y. Loo whose address has somehow gone astray." I would be interested in seeing any comments sent in by your readers on this approach. I hope to write a program in Assembly or Machine language since it offers the one desirable feature and that is speed." ... Thomas A. Fallis, PO Box 76242, Sandy Springs, CA 30328 wrote a letter that appeared in Doug Penrod's 'Newsletter' No. 2. "Since writing that letter," he continues now. "I have formulated exactly what I plan to do with the Chess Data Base I am building. I would like it to be easily interfaced with as many chess programs as possible. To accomplish this, I am asking anyone interested to send me either (a) a copy of their chess program, or (b) a write-up of the program and how it treats positions. The Chess Data Base can be used to retrieve information on any position on file or on general positions of some nature." ... Because of an inability to make proper arrangements in time for the computer show, Chess 4.6 did *not* present a seminar in computer chess as had been hoped. It is a project, though, that may take place at a future show in which more time is available for making proper arrangements. . . .

* * * *

... Chess, one of the few relics of early civilization to survive the ravages and distortions of time, has become the most popular intellectual activity in the world today. Some estimates say that as many as 200,000,000 people throughout the world are currently playing the game of chess in every hamlet of every country, by all ages and at all levels of the social plateaus. Some observers say its fascination lies in the plots designed to overthrow an enemy king. Other observers with more earthy leanings find fasci- in the hopeless struggle of the pawns, symbols of the common people, which plod slowly ahead, one step at a time and with only one goal in mind, to safely reach the last rank of the board. It is not much to ask of life but the pawns rarely achieve their goal. Usually they fall by the wayside – unknown, unheralded and un-mourned. Do players now resent the instrusion of computerized chess into their midst, the way textile workers once resented automation? The answer, obviously, is no. The throbbing enthusiasm that stirs in the soul of the human chess player will never be replaced by a program, a floppy disk, or a casstte. People who come to observe computers wrestle silently on chessboards get as little emotional stimulation as they would if they were watching someone read a travel-folder on public television. 9-ply sub-routine evaluation in a computer-chess program will never elicit a gasp from an awed audience as would a Fischer or a Spassky or a Korchnoi who makes a sudden pompous move with an overlooked rook and announces in a loud, majestic incantation: "Checkmate!" It is the voice of doom, uttered in cloistered halls of chess clubs all over the world. It is highly unlikely that the computer will ever be taught to use that word correctly – to pronounce it with the proper, snap-of-the-fingers fillip. To rephrase a line from a familiar poem: "It's not whether you win or lose – it's *who* you play that counts!" . . .

* * * *

... Thomas C. Ball of 3119 Wynford Drive, Fairfax, VA., 22031 disagrees with Chafitz' method of rating *Boris.* "I have received technical data from Chafitz. I note that Chafitz rates *Boris* as 900-1100 (*earlier models- ed*) but refers to this as Class C instead of D. Strange that someone making and marketing a chess computer would be unfamiliar with chess notation and rating systems. I enjoyed Bobby Fischer's letter in an earlier 'Newsletter' and I agree with Fischer that the reversal of the algebraic notation is unbelievable. By profession I am a physicist, but I'm a layman at machine programming. I have *Microchess* running on my KIM-1 and it plays a remarkable game of chess considering the limited memory. In fact, I have developed a program which I believe would be appropriate for many of your readers. It greatly reduces the running time of Peter Jennings' *Microchess* so that it produces the best move at *Blitz Speed* – approximately 15 seconds per move. It does this by performing a 'check-check' function in an original program that uses less than two pages of additional memory and runs very fast. I would think a lot of people would be interested in getting a copy of this program. In my opinion, the *Microchess* program is the best, inexpensive, readily-available assembly language program for the serious experimenter. On the subject of computers, getting away from chess for a min-

ute, the tape input on my KIM-1 was not working when I took it 'out of the box.' MOS Technology has, as of this writing, yet to respond to my inquiries. In the meantime, I fixed it myself, but speaking as a professional physicist, I think their manuals are very bad. For example, there is no index in the manuals, a feature I would think would be required in any manual. Chapter 6 on 'Expanding Your System,' is unbelievably inadequate. Despite these drawbacks, the KIM-1 appears to be a well made piece of equipment."

* * * *

. . . Bernard W. Klatt of Ottawa, Ontario, describes a chess program available from DEC. " CHEKMO-II, is a chess-playing program for a 4K PDP-8 written by John Comeau of Digital Equipment Corporation. This chess program will also run on nearly all microprocessor systems using the 6100 microprocessor chip. The program plays either White or Black and will accept all classes of legal moves, including castling, both short and long, en passant pawn captures, and pawn promoting moves to any legal promoting piece. The program prints out and accepts moves in Algebraic Notation." Chuck Conley of DECUS adds some information on ordering:

CHEKMO-II is available from the "Digital Equipment Users Society" (DECUS) on either binary paper tape or on an OS/8 DECtape. The media prices are as follows:

- Paper tape – $6 (including source listing)
- DECtape – $32 (including source listing)
- DECtop floppy disk, also OS/8 compatible – $27

The DECUS catalog number is 8-822 CHEKMO-II: Chess program. Also available from DECUS is a chess program written in BASIC-8 which can run on a PDP/8, a DEC/10 or a Hewlitt Packard. This program which has a DECUS catalog number of BASIC 8-104 sells for $2.00. It is also known as "QChess, Quigley's Algebraic chess program written in BASIC." DECUS is the acronym derived from Digital Equipment Customer Users Society, a world-wide, non-profit, Digital-sponsored society with 30,000 members in four chapters: U.S., Canada, Europe and Australia. Address for inquiring about membership or for ordering above-mentioned programs is: DECUS Library, Digital Equipment Corp., Building MR 2-3/E55, 1 Iron Way, Marlborough, MA 01752.

Writing a Chess Program

This complete dissertation by Mike Valenti on how to write a computer chess program is presented in monthly sections as a guide to those wishing to write their own programs. Although designed to be run on a large computer, this program with proper modifications can serve also as a model in writing a chess program for smaller memory-systems – even the microcomputer. This program is written in BPL (modified XPL), but it can be written in other languages as well – with proper transitions.

Basic Program Structure

The basic flow of control in the program is shown in figure below. The basic layout and strategy are similar to most other chess playing programs. There are a few differences, however. The legal move generation routines were incorporated into the data structure generation and plausible move selection routines. This means that the plausible move analysis is done even for non-computer moves. But this takes only 0.16 seconds of CPU time on an IBM 370/165 and makes the program simpler in structure. Also, the look-ahead strategy was changed so that the best move is selected on the basis of the best series of moves, or best path in the tree, as opposed to selecting the best final position as most other programs.

Referring to the figure, the INITIALIZE-DATA-STRUCTURE routine generates legal moves for pieces and fills out the data structure for the current position. The PLAUSIBILITY-ANALYSIS routine takes this information and generates more detailed information pertaining to all of the legal moves from this position, such as pinned-piece information and the locations of open files and passed pawns. This information is used by the HEURISTICS routine which is called for each legal move. These heuristics give scores to each move passed to them, and the PLAUSIBILITY-ANALYSIS routine orders the moves according to these scores.

The best moves are now given more detailed consideration by means of look-ahead. The LOOK-AHEAD routine uses INITIALIZE-DATA-STRUCTURE and PLAUSIBILITY-ANALYSIS to generate responses to each of the first moves considered. A game tree is formed by repeating this process and modifying the moves' scores at the top level of the tree. These new scores are referred to as the "backed-up values" and the computer selects the move with the best backed-up value (except in avoiding repeating positions if it is ahead in material).

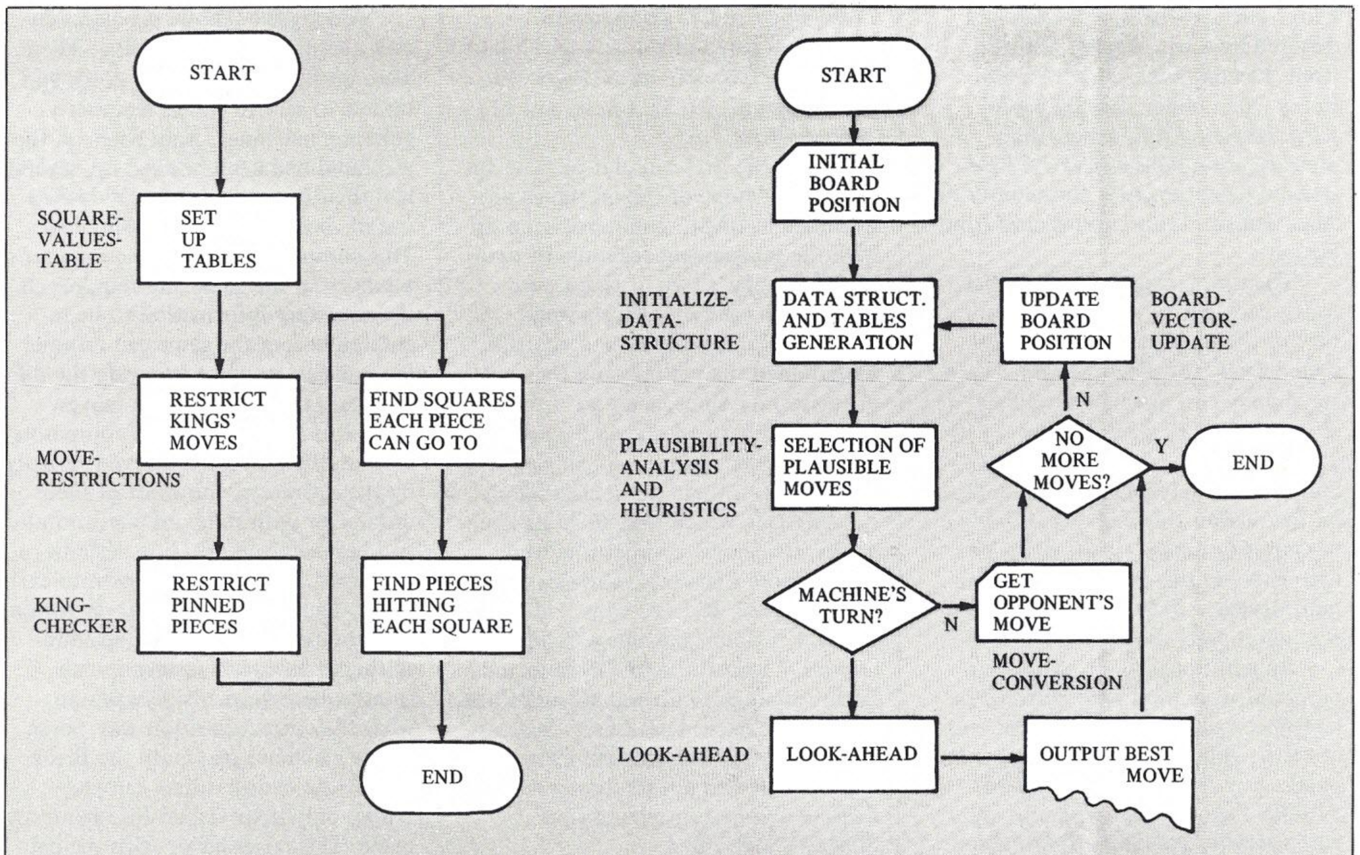
All moves by the computer and its opponent pass through a general move scanning routine, MOVE-CONVERSION, in order to check their validity against the data structure. Any commands that are entered are also processed by the MOVE-CONVERSION routine.

The BOARD-VECTOR-UPDATE routine is supplied with the form and to-squares of a legal move and modifies the board vector accordingly. It also returns a value that indicates the type of move made, for instance, a capturing, non-capturing or king-side castling move, etc.

Data Structure and Tables Generation

The INITIALIZE-DATA-STRUCTURE routine generates legal moves for all of the pieces on both sides and places information in tables for later use in the plausible move selection for the side to move. The second figure shows the way in which information is gathered. The chess position data structure is filled out as the information is gathered. As the data for each piece are being generated, information about what squares are being hit by these pieces is saved in an array for later reference when the data for each square are generated.

This array is a 64 (number of squares) by 12 (maximum number of pieces hitting any square) that typically contains very little data relative to



INITIALIZE DATA STRUCTURE FLOW CHART

OVERALL FLOW OF CONTROL

its size. This information is therefore saved in list form in the chess position data structure previously discussed. An additional array of 64 by 25, again very sparse, is used to save pieces that are indirectly hitting a given square. These indirect hits are with either one or two pieces intervening and pairs or triples of piece numbers are saved for inclusion in the chess position data structure.

The square values table is set up and some information is placed in the location table (described in detail in the section to follow) and information is saved in the MOVE-RESTRICTIONS routine (see second figure in this article) regarding pieces pinned against the kings. The pin information is required in attack evaluations in checking if a piece that has been unpinned in the exchange, can now legally enter in the exchange.

The above data are then passed to the PLAUSIBILITY-ANALYSIS routine for plausible move selection and ordering. The look-ahead routine (described later) uses INITIALIZE-DATA-STRUCTURE and PLAUSIBILITY-ANALYSIS for each new position encountered in the game tree. The chess position data structure (using about 4000 bytes of controlled storage) for the initial position being evaluated in look-ahead is retained. Therefore, 8000 bytes of controlled storage are required for these. (To be continued.)

More on San Jose

. . . The following story appeared in CHESS VOICE, Official Publication of the North California Chess Association. It is the work of John Larkins, Editor of CHESS VOICE and Larry Wagner, director of the San Jose Tournament:

"For centuries,men have competed with each other over the chessboard. Then, with the advent of chess-playing computers, they started to play machines. Now the machines are playing each other. There have been national and international computer chess championships for several years. (The current world champion computer is America's *Chess 4.6,* which is now consistently playing at an expert level.) But these tourneys involve an assembly of computer terminals each connected by telephone hookups to remote machines, some thousands of miles away, whose value could be measured in millions of dollars. (It costs $38/second just to run the *Chess 4.6* program.)

"The most recent development in this field is the appearance of a number of chess-playing microcomputers. These are small, self-contained, relatively inexpensive machines that play chess at about Class D or E strength. Some are marketed commercially for the sole purpose of playing chess; others are programs that can be used with personal computers designed to carry out a variety of other tasks as well.

"The world's first microcomputer chess tournament (machines vs. machines) was held March 3-5 in San Jose at the 2nd West Coast Computer

Faire. The event was organized and directed by Larry Wagner, assisted from the computer world by John Keary, Alan Miller, Ian Shepperd, Larry Kaplan, Craig Asher, Brad Stewart, John Mills and Daryl Elder, assisted from the chess community by Alan Benson, John Larkins, and John Spargo.

"The tournament had 11 participants, each with a distinct program. Five of the machines have been designed solely for playing chess. Three are already commercially available: 1) *Boris* ($300), 2) *CompuChess* ($160-220), and 3) *Chess Challenger* ($220-280). A fourth is scheduled to appear on the market this Fall – 4) *Commodore Chessmate* ($150-225). The fifth entry in this group was a home-built device – 5) *Steve Stuart's Metal Box,* with parts that cost a mere $85.

"An additional six chess-playing programs were used with personal or hobby computers. The programs cost in the neighborhood of $20; the computers vary from about $2,000 to $7,000. Two of these programs are commercially available: 6) *Processor Technology* and 7) *Compucolor.* Four others are individually-developed programs: 8) *SARGON,* 9) *Mark Watson's Program,* 10) *SD Chess,* and 11) *Tenberg Basic.*

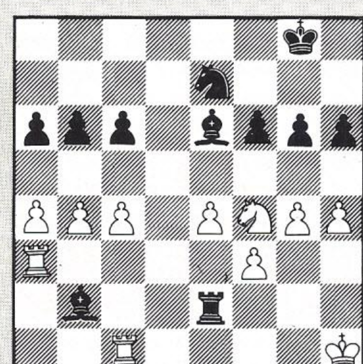
"After the first round it became apparent that some of the machines programmed in BASIC could not keep up with the tournament schedule of two games per day. (Two of them were paired with each other for a single nine-hour game.) Unfortunately, this undermined the pairings and the clarity of the results, since, after six "rounds", some machines had played 5 games, some 4 games, and one – a single game.

"However, there was a clear winner – SARGON, which won all of its five games. Its nearest competitors were Commodore Chessmate (2 wins, 2 draws, 1 loss), Boris (2 wins, 1 draw, 1 loss, 1 forfeit), and Chess Challenger (3 wins, 2 losses). SARGON defeated both Commodore Chessmate and Chess Challenger. Boris beat Chess Challenger, but lost to Commodore Chessmate. Chess Challenger got its three wins from two Class C machines plus Stewart's Metal Box.

"Among the unique problems encountered were the following: There were insufficient electrical cords and outlets to service all the electricity-gobbling machines. And, when all the machines had been hooked up, several had their cords inadvertantly kicked out of their sockets by passing feet. This completely erases the computer's memory of the game and requires all the necessary information to be re-entered before the game can resume. Particularly troublesome were the difficulties in communicating moves. Since it is hard to shift the computers around, the moves are relayed verbally by the operators. But most of them are unfamiliar with standard chess notation and have evolved notation systems of their own, which are unknown to each other. Thus many errors in translation and notation were made – again resulting in having to restart several games where neither side was sure what the correct position was. Some of the machines play only the Black side of the board; others can play White, only if special arrangements are made. This raises havoc with the pair-

White – **Processor Technology** Black – **Commodore Chess Mate**

1. e4	e5
2. d4	ed:
3. Qd4:	Nc6
4. Qd5(a)	Bb4+
5. Nc3	Nge7



Position after Black's 27th move. Both players have achieved a forking position. Who profits from the resulting exchange? Read on.

6. Qd3	O-O
7. Nf3	d6 (b)
8. Bf4	Ne5 (c)
9. Be5:	de:
10. Qd8:	Rd8:
11. Ne5:	Rd4
12. f3	Be6
13. Be2	Rad8
14. O-O	Rd2
15. Bd3	Rd4
16. Nb1 (d)	Bc5 (e)
17. Nd2: (f)	Rd3:+
18. Kh1	Rd2:
19. Rfc1	f6
20. Nd3	Bd4
21. h4	c6
22. g4	a6
23. a4	g6
24. Ra3	b6
25. b4	h6
26. c4	Re2
27. Nf4(g)	Bb2
28. Ne2:	Ba3:
29. Rc3	Bb4:
30. Rc2	a5
31. Nd4	Bf7
32. f4	c5
33. Nf3	Nc6
34. Kg2	Nd4
35. Nd4: (h)	cd:
36. Kf3	d3
37. Rc1	d2
38. Rc2 (i)	d1=Q+
39. Re2	Bc4:
40. h5	Qe2:+
41. Kg3	Be1+
42. Resigns	

Annotations by Alan Benson

(a) This opening theory only considers two main continuations here: 4. Qe3 and 4. Qa4.
(b) Why not 7 . . . d5 with the initiative?
(c) Loses a pawn.
(d) A fine move winning the exchange.
(e) Setting a little trap.
(f) Falls right into it! Instead 17. Kh1 keeps the material edge.
(g) The "knight fork" to which black has a clever resource.
(h) This makes it easy for black. Better was 35. Rc1 but after 35 . . . Be8 36. Ra1, Nc2 37. Ra2, Ne3+ followed by 38 . . . Nc4: black is winning.
(i) A surprising move. It's lost in any event as after 38. Rd1, Bc4: followed by 39 . . . Bb3 winning the rook.

ings – as did the occasional unavailability of operators who had other things to do at the Computer Faire.

"Compared to typical human tournament players, all the microcomputers tend to lack the kind of killer instinct needed to finish off an opponent once he (it) is on the ropes. Ten of the 22 games played at San Jose had to be abandoned midstream. Four ended when one machine was not able to make the time control: four more ended by adjudication after four hours of play; and two games were declared a draw "by lock-up" when one machine went into a cycle of repetitive moves. (The 22 completed games were all mates. None of the microcomputers is programmed to offer a draw or resign.)

"Special rules were set up to deal with the lock-up problem. The typical lock-up will occur in a rook and pawn endgame where one side has a won game but nevertheless insists on an endless series of rook checks. (A lock-up is like a hiccup.) To prevent these games from ending in a draw, the operator was allowed to increase the look-ahead capacity of the locked-up machine, with the hope that it could then see it wasn't getting anywhere, and try something different.

"Most chess-playing microcomputers have a variable look-ahead capacity. But they pay the price for deeper tree searches in greatly decreased response time. CompuChess, for example, is good at solving mate-in-two problems with its Level 6 mode – but it can take up to two days per move! Most of the machines at San Jose were playing below their maximum theoretical look-ahead capacity in order to meet the time control. CompuChess played at Level III; Boris at 2 minutes per move. In two cases, even with increased look-ahead, the machine continued to repeat itself – these are the two draws by Commodore Chessmate. In a number of other cases, the hiccup response was bypassed, allowing the machine with the extra material to survive to the four-hour time limit, where it could win by adjudication.

"Often the microcomputers play chess just like people do. But every once in a while, one gets the sense of a definitely alien intelligence at work. An amusing example occurred in the game between Processor and Commodore. After Black's 20th move they had arrived at a quiet but puzzling middle game position. Neither side could come up with a game plan, and, in the absence of any obvious tactical shots, both began an almost random pushing of wing pawns. Curiously, Processor appeared to have been taught to push its pawns two squares, when in doubt, while Commodore had been told to push them one square. (See moves 21 to 26 on preceding page.)

"Since the San Jose tournament, SARGON has undergone total revision. It now sees ahead twice as far and is called SARGON II. (The program is available for $15 from the Spracklins.) Steve Stewart, too, has a new program and a new metal box. Mark Watson is translating his BASIC program into assembly language." (Subscriptions to CHESS VOICE, or information, can be obtained by writing to Editor John Larkins, CHESS VOICE, 5804 Ocean View Drive, Oakland, CA 94618.)

The Roumanian match

. . . When we last left Felix C-256 and his chess program (ASTRO 64) the new Roumanian computer had reached the 9th move of his public match in a Roumanian newspaper. Readers of Bucharest's *Magazinul* had Felix hanging on for dear life by the finger tips and the natives were savagely stomping on those digits attempting to dislodge Felix. Did they succeed? Was Felix forced to fling himself from the cliffs to his doom? Read on, dear readers. Our Roumanian correspondent U. Valureanu, again warns us to be wary. Like all cats, he says, Felix is clever and should be able to get out of any situation. Position at the end of the 13th move is shown in accompanying diagram and the next four official moves between Felix and the readers of *Magazinul* were:

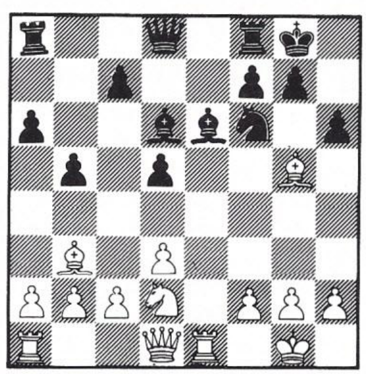
White – *Roumanian Readers*

Black – *ASTRO 64*

10. RxN+	B-e6
11. B-g5	Bf8-d6
12. R-el	O-O
13. N-d2	h7-h6

How did your four moves compare to our Roumanian brothers' and sisters"? Think you did better? Roumanian readers would like to see how well their American counterparts are doing, so if you send us your moves #10 to #13 we will relay them to Bucharest for their enlightenment. Meanwhile, the newspaper readers seem to have a strong attack still mounting against the computer. Enter your own next four moves here and watch as the international drama continues to unfold:

14.	. . . . . .	16.	. . . . . .
15.	. . . . . .	17.	. . . . . .



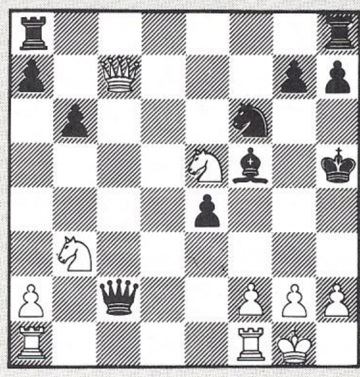
Levy outwits the computer

. . . . . . During ACM's North American computer chess tourney, held at Seattle in Oct. 1977, David Levy played all 12 computer-programs at the same time. The only machine able to beat him was his arch enemy at the chessboard, *Chess 4.6*. (Whether *Chess 4.6* can duplicate this achievement on a one-to-one match has been a debatable point in the chess world for many years.) One of the easiest games Levy played was the one against *TYRO,* a program from the University of Southern California. *TYRO* was run on a giant PDP 10KL computer with a memory size of 270K. It could examine 10,000 positions per move. This ability of all computers, to examine massive numbers of chessboard positions during a game is the basis of the argument that a computer's ability to consider 200,000 positions is a useless talent. A grandmaster considers, at the most, perhaps 5,000 positions. Therefore, when pitted against a computer that considers 200,000 the grandmaster, one might think, should be at a

disadvantage. However, arguments in favor of the grandmaster are that the positions he considers are only valid ones. He does not bother with positions which he quickly recognizes as being hopeless. The computer, on the other hand, must examine *all* possibilities so that it can evaluate the terminal branches of its tree search and arrive at the best comparative value for the next move. Current research is focusing on recognizing chessboard patterns.

White: *David Levy* Black: *TYRO*

1. e4	e5
2. Nf3(a)	Nc6
3. d4	ed:
4. c3	dc:(b)
5. Bc4	Bc5
6. Bf7:+(c)	Kf7:
7. Qd5+	Kf8



Position at move 23. White is about to snatch off Black's Queen after trapping it in a series of brilliant moves.

8. Qc5:+(d)	Qe7
9. Qc3:	d5
10. O-0	de:
11. Be3	Qf6
12. Bc5+	Kf7
13. Qb3+	Kg6
14. Nfd2	Qe5
15. Qc2	Bf5
16. Be3	Nd4
17. Bd4:	Qd4:
18. Qc7:	Qb2:
19. Nb3	b6
20. Nld2	Nf6
21. Nc4(e)	Qc2
22. Ne5+(f)	Kh5
23. Qc2:	Rac8
24. Qe2+	Kh4
25. g3+	Kh3
26. f3	ef:
27. Nf3:	Ne4
28. Ng5+	Ng5:
29. Qh5 mate	

Annotations

(a) This was Levy's favorite opening during all his exhibition matches against the computers – bringing out one of his Knights at the second move. This aggresive opening probably startled the computers who were accustomed to more conservative play.
(b) Moving the pawn out of White's Queen file gives Levy an early advantage. It provides his Queen with practically an open ballroom in which she can dance about at will.
(c) This sacrifice by Levy throws such a fright into Black that after this play, its calibre of performance is greatly reduced.
(d) These three successive checks, comparable to a boxers sharp jabs before a knockout, disrupts *TYRO*'s game and weakens him for the big blows about to come.
(e) This is *TYRO*'s major blunder. It locks his Queen in a hopeless position. Black's only move from this position is e2 which, if not a life saver, would have, at least delayed the ending.
(f) The brilliant sequence of moves seals *TYRO*'s doom. The check exposes his Queen to free capture. Black is helpless.

A micro tournament

. . . Don Gerue and Russ McNeil, of Santa Barbara, are staging a microcomputer-chess match on their own premises. They have on hand: Boris, Chess Challenger, Microchess (in 3 versions) the new Atari program, and SARGON (winner of the first microcomputer chess tournament in March, '78). The software programs will be run on a Heath, a TRS-80 and a PET, the three micros available to Don and Russ. Actual games are played on weekends, in the presence of interested friends who act as judges. Information on the game results will appear in December issue of PERSONAL COMPUTING. The games will be run in a double round-robin pattern; that is, the programs will first play white and then black, against each other. More information on this exciting tournament can be obtained from Don Gerue, Tel 805-682-1270. Or he can be reached by mail at 3667 Montalzo Way, Santa Barbara, CA 93105.

Checker challenge

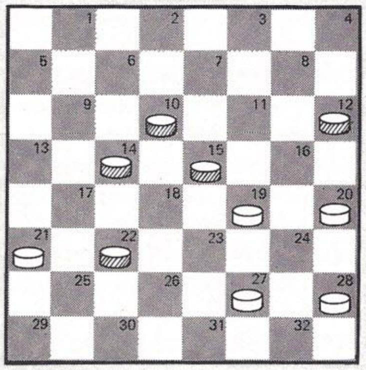
. . . Burke Grandjean of the American Checker Federation announces that world's checker champ, Dr. Marion F. Tinseley, of Florida A&M, has a wager of $5000 that he will be able to beat any computer at checkers within a five-year period. The money, says Burke, will be put in escrow (unlike Levy's chess bet that was supposedly only a "gentleman's agreement.") Put-up-or-shut-up being the classic battle cry for all gaming contestants, The American Checker Federation "has put up." The question now emerges: how many challengers are there around or are likely to emerge? Checkers is known to be a much easier game than chess for microcomputer programming. For further information contact American Checker Federation, 3475 Belmont Ave., Baton Rouge, LA 70808.

Duke vs Lowder

. . . In May of 1977, Duke University's Computer-Checker program, originally authored by Eric Jensen with later help from Tom Truscott and others, took on Elbert Lowder. Lowder recently placed second in the Champion Division of the 1978 National Checker Tournament, which was held in Kentucky. The games between Lowder and the Duke program were played in a quiet room with Lowder allowed "unlimited time per move and with the Duke program averaging 80 seconds per move. Lowder played somewhat more rapidly than the program. Occasionally, though, he pondered a move for as long as five minutes. Following is one of five games played at that time. (Annotations by RL Fortman, Games Editor of American Checker Federation):

Black	White
1. 10-15	21-17
2. 7-10(A)	17-14(B)
3. 10-17	22-13
4. 11-16(C)	23-19(D)
5. 16-23	26-10
6. 6-15	13-6
7. 2-9?(E)	25-22
8. 8-11	27-23

Black – *Duke Computer* White – *Elbert Lowder*
(Black Moves First)



Position after Black's 21st move upon which Lowder concedes. In the continuation of the game, Dick Fortman notes the following moves: 27-24, 22-26, 20-16, 26-31, 16-11, 31-27, etc. which leads finally to a loss of one of White's pieces. Further continuation leads to another loss and no further hope for Lowder.

Annotations by R.L. Fortman

(A) This designated "3-move opening" was programmed into the computer. The opening is known as the "Octopus," and is one of the weakest of the 142 approved openings by the ACF.
(B) Mr. Lowder is quite familiar with the correct attack as used here, (which improved the computer's selection of 17-13 in a later game.)
(C) Since it is unnatural (permitting the subsequent break-up) one might surmise it had been previously programmed, but Note E may contradict this opinion! More natural moves, but losing, are 9-14 and 3-7.
(D) White has a variety of powerful attacks, but the text is one of the best; breaking the Black double-corner side.
(E) It would seem that many computer programs favor the side captures, instead of the better center options. This loses, with 1-10 instead extensively analyzed to a thin draw.
(F) Satisfactory, although 22-17, 11-16 and the 24-19 exchange has been published to a White win by former ACF editor Lee Munger, or Port Charlotte, Florida.
(G) Here, 29-15 would seem to win after 9-14 and again the 24-19 exchange.
(H) A questionable venture, permitting Black to bind. Instead, 24-20 (as 17-13 allows 11-16, 13-6 and 15-9 etc.) leaves White in a strong position.
(I) Here, 17-14, 10-17, 23-19 might be attempted, then 17-22, 19-10, 22-29, and 10-7 with drawing chances.
(J) White is now in deep trouble.
(K) Finely played and cements the win. The 1-5 or 1-6 alternatives would permit draws.
(L) White, temporarily a piece ahead, has no satisfactory waiting moves.
(M) With the vain hope that the pitch in following notation (N) would not be discovered.
(N) Again finely played as the discovery is made.
(O) Although the computer made a serious error earlier (see notation E) its later play was most exemplary after White passed up the wins. The computer's late mid-game play was a great improvement over its efforts vs. the Stanford computer games of 1976.

Black	White
9. 4-8	32-27(F)
10. 3-7	22-17(G)
11. 7-10	29-25?(H)
12. *9-13	25-22(I)
13. 5-9	30-25(J)
14. 9-14	25-21
15. 15-18(K)	22-6
16. 1-10	23-19(L)
17. 13-22	19-16(M)
18. 11-20	24-19
19. 20-24(N)	27-20
20. 8-11	31-27
21. 11-15(O)	

BLACK (DUKE) wins

Historical notes

(Brief extracts and comments on computer-chess development from 1941 to present date. The following is from "A Five-Year Plan for Automatic Chess" by Dr. I.J. Good of Virginia Polytechnic Institute, Blackburg. It was written in 1967 and is reprinted here with his permission.)

The nature of complex games

There is one principle of chess that is so old that early theoreticians probably did not bother to mention it. It is certainly as old as the game of Go, at least 3000 years, and is typical of all complex games, although stochastic games have additional features. It is the habit of thinking 'if I go there, then he might go there or there, and, in the first case, I might go there, etc.'. In other words, analysis often follows the branches of a tree, with evaluations at the endpoints, followed by back-tracking or iterative minimaxing.

The notion of a game tree applies of course much more generally than to chess. With regard to the definition of the endpoints, it is sometimes said that they should be quiescent positions, but provided that the outcome of the game is clear enough, an endpoint can also rationally be a turbulent position. For this reason, it is convenient to introduce a notion called 'agitation', which is a modification of 'turbulence' taking into account the probability that the result is clear-cut.

Let us suppose that we can measure, for any position, π, both its *turbulence* and also the *superficial probabilities* that the player can win, draw, or lose. p_W, p_D, and p_L. (By a 'superficial probability' I mean a probability based on some evaluation function, without any forward analysis. Perhaps 'surface probability' would be a better term.) Then the decision of whether to regard that position, π, as an endpoint, when analysing a position, π_o, depends on the following considerations:

(a) The *depth* (i.e., the number of moves ahead) of the position π from the current position, π_o. The smaller this probability, $P(\pi | \pi_o)$, or, less accurately, the greater the depth, the more rational it is to treat π as an endpoint of the analysis tree for π_o.

(b) The lower the turbulence of π, the more prepared we should be to treat it as an endpoint, other things being equal.

(c) The more obviously the outcome of the game is decided at π, the more prepared we should be to treat π as an endpoint, other things being equal.

(d) The larger the analysis tree as a whole, as far as we can judge, the more

prepared we should be to treat π as an endpoint, other things being equal.

(e) The less time we have left on our clock, the more we should be prepared to treat π as an endpoint, other things being equal.

(f) The less time our opponent has left on his clock, the more prepared we should be to treat π as an endpoint, other things being equal. (This advice is double-edged!)

In other words, we should define a certain function of six variables, monotonic in each separately, and should treat π as an endpoint if this function exceeds some threshold.

Notice the following philosophical point. The superficial probabilities are not strict logical probabilities, since they are based on a suppression of a logical argument, namely a forward analysis. Since we are forced to make use of superficial probabilities, there is bound to be an element of luck in chess. From the point of view of strict logic, chess is a game of pure skill, but it is a physical impossibility to avoid some degree of chance. Just as in mathematics we must make use of probabilities based on incomplete analysis. When trying to prove a mathematical theorem, it is necessary to formulate subgoals, and to estimate superficial probabilities for them, in order to find an appropriate strategy. Theorem-proving resembles chess playing in that we have an objective and an analysis tree, or graph, but differs in that a superficial expected pay-off replaces the iterated minimax. The minimax idea can come in if we are trying to prove a theorem and we imagine that we have an opponent who wishes to disprove it. The value of our game is 1 if the theorem is true and -1 of it is false. In proof trees the 'and's correspond to moves of the opponent, since we must allow for both branches, whereas the 'or's correspond to our own moves. The minimax (strictly maximin) value of the tree tells us whether the theorem is true, and, if we allow for superficial probabilities at the endpoints of the tree, the minimax value is the superficial probability of the theorem. The programming of complex games will also exemplify an aspect of practical decisions, since here too it is typical that we are forced to make use of superficial probabilities.

If our object is to learn as much as possible about learning, we could try to program a machine to learn the game without even telling it the rules. This is how Capablanca learned chess: by watching his father and uncle playing. Incidentally he won the first game he played at the age of 6.

Next, we could give the rules to the machine and let it learn the principles for itself. Here again the machine would need to be able to formulate a wide class of hypotheses.

Next, we could give the machine a limited class of principles and allow it to work out its own evaluation functions in terms of these principles in the light of its experience. The experience could either be the playing of games, or, more efficiently, we could tell the machine which moves in various positions were good or bad, and how good or bad. The simplest form of evaluation function would be linear, with coefficients that were allowed to be positive, negative or zero. The optimisation of these coefficients constitutes a primitive form of learning, but I think most people would be reluctant to call it 'concept formation'. If however non-linear polynomials are permitted, or equivalently if logical combinations of the principles can be formed and incorporated into the linear function, then concept formation becomes possible. For example, a machine might have discovered that two bishops are usually better than bishop and knight, if it had been programmed to examine quadratic evaluation functions. After this had been established, it might have gone gone on to modify the concept by saying that it was true in open positions, but not in closed positions. It might have discovered the whole of this principle in one step if it had been allowed to use cubic evaluation functions, but this would be a more expensive way of making the discovery. It seems reasonable to say that the use of quadratic evaluation functions provides the first step in automatic creativity or concept formation.

"If our object is to learn as much as possible about learning, we could try to program a machine to learn the game without even telling it the rules. This is how Capablanca learned chess: by watching his father and uncle playing. Incidentally, he won the first game he played – at the age of six!"

Humans usually think this way. First they give relative weights to features, then they look for interactions between pairs of these features. When they find a pair of features that need to be taken together, they define this as a new feature. Thus high-order interactions are often discovered piecemeal, but of course this is not always possible. The discovery of more intricate concepts will require the generation of a large number of propositions, and the order in which they should be generated provides a most interesting search problem.

We could try to write the program with man-machine synergy in mind from the start. We would aim to teach the machine how to play chess, and also to learn something about chess from the machine. Our long-term aim would be to cooperate with the machine in games against grand-masters, against other machines, and against other synergistic combines.

We could go all out to make the machine a good chess player, by putting as much as we can into the low-level program, and without relying on more machine learning than is implied in the optimisation of linear forms.

Of these various aims, which are not mutually exclusive, the one which mentions Capablanca is the most exciting, but the paragraph just preceding this one would be easier. I think a reasonable approach is to aim at this last one at first, in the hope of getting a good chess program, but always holding the other aims in mind.

LOOKING FOR THE NEW KENTUCKY FRIED CHICKEN OR McDONALD'S? JUST OPEN YOUR EYES!

Back in the fifties, if someone had suggested you invest in a hamburger stand called McDonald's or a chicken store run by Colonel Sanders, you probably would have laughed. Most of us did. The few who didn't, and invested in KFC or Big Mac are millionaires today. They enjoy "finger lickin' good" profits and "have it all done" for them.

The whole trick to investing in your own business is to **keep your eyes open for something like a KFC or McDonald's**. A business that (1) requires a **small investment** that can be recouped quickly, (2) has an **enormous profit margin**, and (3) has great growing **consumer acceptance.**

There is such a business.

The business is computer portaits, and it's one of the hottest, most profitable new ideas around. International Entrepeneur's Magazine stated that there are locations that are currently grossing **from $2,000 to $4,000 a week**. Imagine, grossing up to $4,000 a week from a small investment that gives you **your own high volume, all cash business.** No franchise fees or royalty payments, **all the money is yours**.

Computer Amusement Systems, Inc., (CASI) of 11 West 20th Street in New York City, has taken today's hot trends—T.V., computers, and instant pictures and combined them to produce a computer portrait system that is high in quality, low in price, portable and **requires absolutely no photo or technical experience.**

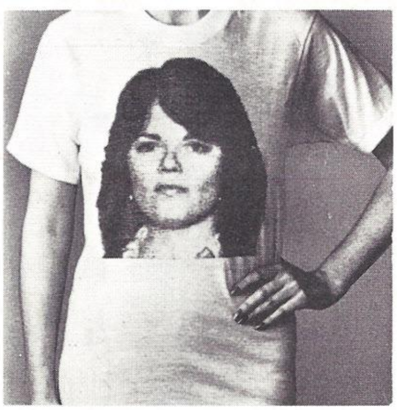
Easy to operate and easy to move, **the portable CASI system can be set up anywhere:** malls; flea markets; shopping centers; conventions; rock concerts, anywhere with high pedestrian traffic and just a little floor space. This **instant traffic stopper** will make a computer portrait in just fifty-five seconds! The picture is first seen on a T.V. screen, then dramatically printed before your eyes.

And there's more. You can transfer the portraits instantly to **many high mark-up, big profit items**—tote bags, T-shirts, calendars, puzzles, dart games—whatever the latest trend might be. CASI supplies the wholesale sources for everything you need to be in on the profits.

And there's more. Like special options that allow personalized messages to be printed right on the computer portrait. Or programs that will print out personal bio-rhythm charts in seconds. **All big moneymakers**.

So open your eyes to the most dynamic, profit making opportunity of the year. **Perfect for part time, full time, family operation or absentee management**, CASI is your ticket to success. Start putting money in your pocket today with CASI computer portrait systems. Don't send money just write for details. We'll send them right away.

Before you make an important decision in this exciting new business, know who you're buying from. We suggest you ask your banker about us.



YES! I WANT TO BE THE FIRST IN TOWN TO CASH IN ON COMPUTER PORTRAITS. PLEASE RUSH DETAILS

Name ______

Address ______

City ______ State ______ Zip ______

Home Phone ______ Business Phone ______

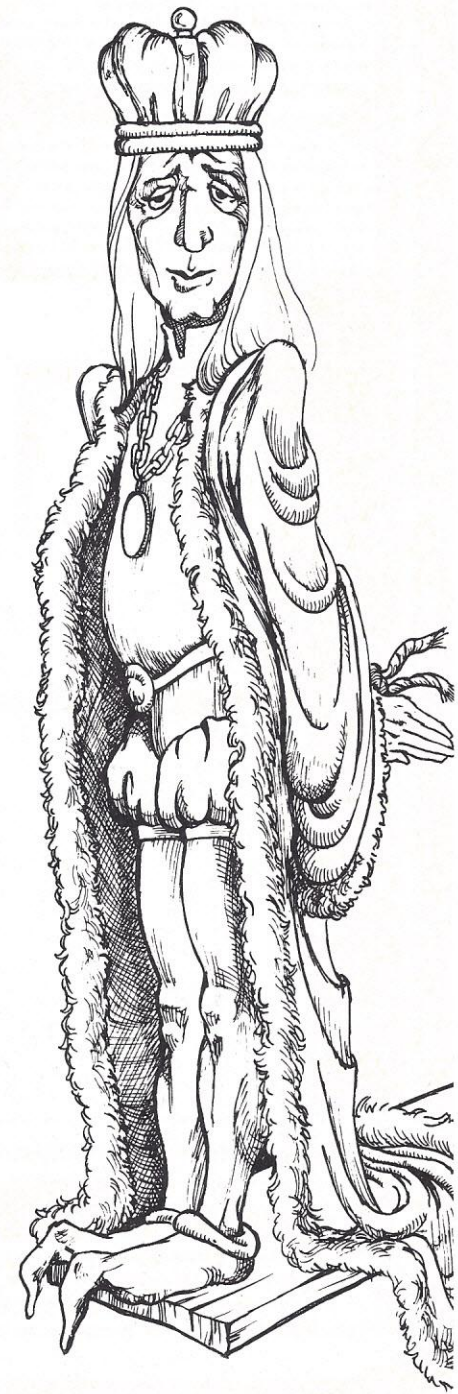
CASI

COMPUTER AMUSEMENT SYSTEMS INC.
11 West 20 Street New York, N.Y. 10011, Dept. 10
(212) 929-8355

P

CIRCLE 9

DEALING WITH ADVERSARIES



Patrick Henry Winston

This extract is from an excellent authoritative textbook titled Artificial Intelligence by *Winston, © 1977. It is reprinted with permission from the publishers. Addison Wesley, Reading, MA 01867. Patrick Henry Winston is Associate Professor of Electrical Engineering at M.I.T., Cambridge, and is director of the Artificial Intelligence Laboratory at the institute.*

A unique sort of computer search comes up because of work on games, a subject that remains one of great unrest. For computer games like chess, checkers, and the oriental game, Go, some people feel that hardly any knowledge is useful beyond whatever is needed to look ahead through many rounds of move and counter-move, imagining the likely consequences of each possible play. Others claim that brilliant strategic play is beyond the capability of approaches based solely on look ahead – much more is required from the computer in the direction of understanding and reacting to various sorts of position patterns.

Thought and practice so far have favored the look-ahead focus, perhaps because the subject of search is relatively tractable. The nodes in the computer's search tree naturally represent board configurations, and they are linked by way of moves that transform one situation into another. (See Fig. 1) Of course there is a new twist in that the decisions are made by two people, acting as adversaries, each making a decision in turn.

Real games involve quite a few more than the two moves illustrated at any point in the game. The average is given by a statistic known as the branching factor.

■ The branching factor is the average number of legal moves possible at a given position in a game.

It is the combination of branching factor and the average number of moves

in a game that generally demonstrates the absurdity of trying to look ahead to the end of the game. In chess, for example, a reasonable estimate for branching factor is about 35 and typically each player makes 50 moves. Certainly 35^{100} is too big.

If only there were some very good way to rank the members of a set of board situations, it would be a simple matter to use the move which leads to the best situation that can be reached by one move. Unfortunately, no such situation-ranking formula exists. When board situations differ obviously, then some simple measure like piece count can supply a rough guide as to quality, but depending on such a measure to rank the available moves from a given situation produces poor results. Some other strategy is needed.

The obvious generalization is to use a board situation evaluator not immediately, but after play has extended through one or more rounds of move and counter move. This cannot be pursued too far since combinatorial explosion soon leads to unthinkable numbers, but if the development terminates at some reasonable depth, then perhaps the terminal situations can be compared, yielding a basis for more selection. Of course, the underlying presumption of this approach is the notion that the merit of a situation clarifies as it is pursued by the computer and that the look-ahead process can extend far enough that even rough board-evaluation formulas may be satisfactory This presumption is hotly debated.

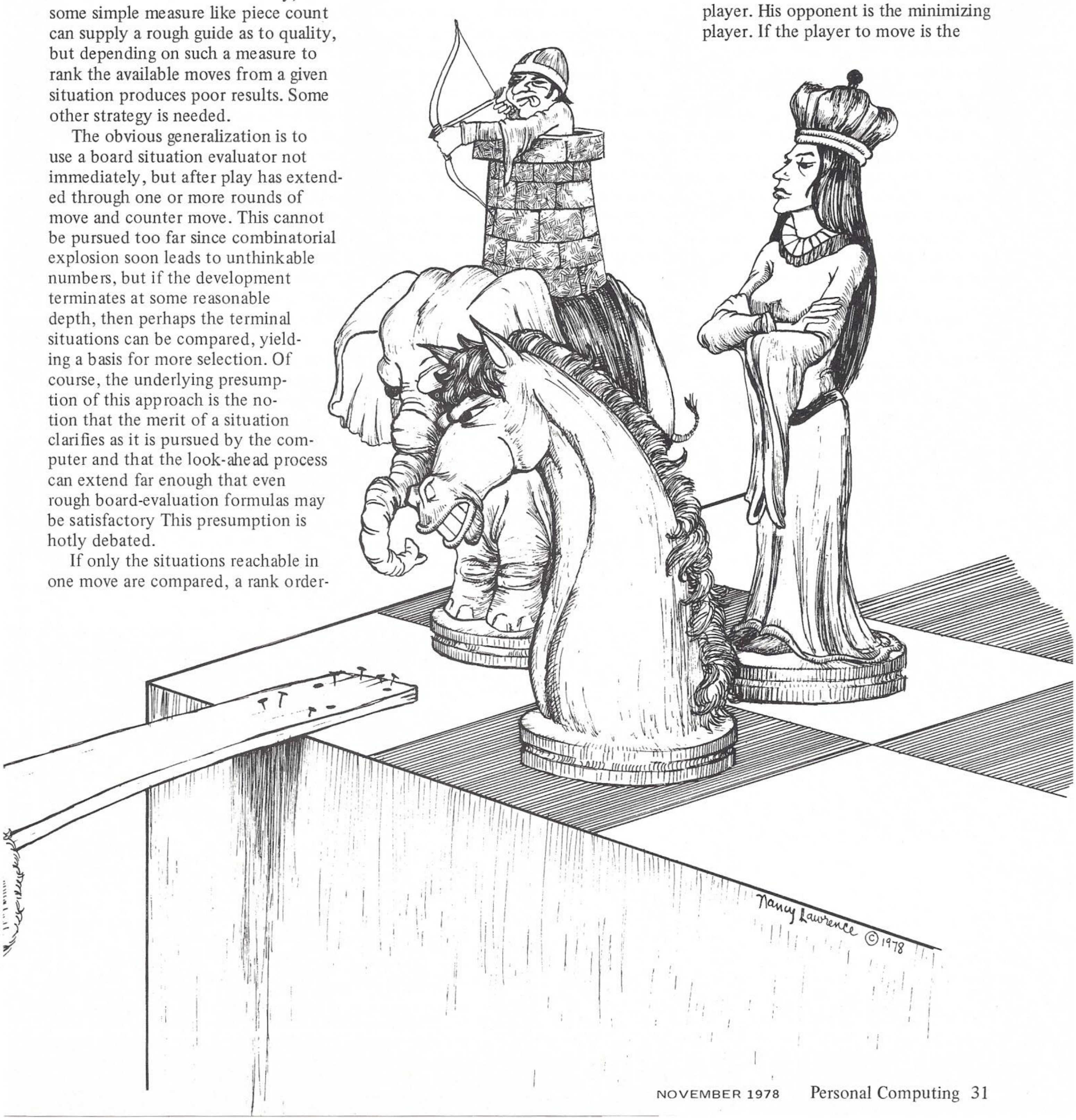
If only the situations reachable in one move are compared, a rank ordering of those situations will do. Unfortunately this is inadequate in the context of look ahead play because an absolute rather than relative standard is needed. The tendency has been to map all considerations into a single, overall quality number. This is forced by the desire to use the simple search strategies we are about to consider, but mapping overall quality into a single number has a serious defect.

■ A number can say nothing about how it was determined. It is a poor summary mechanism, often forced on us by what little we know how to do. Too bad.

Positive numbers, by convention, indicate favor to one player, and negative numbers, to the other. The degree of favor goes with the absolute value of the number.

■ The process of determining the quality number is called static evaluation. At the end of a limited exploration of move possibilities one finds *static evaluation scores* produced by the *static evaluator.*

The player working toward positive numbers is called the maximizing player. His opponent is the minimizing player. If the player to move is the



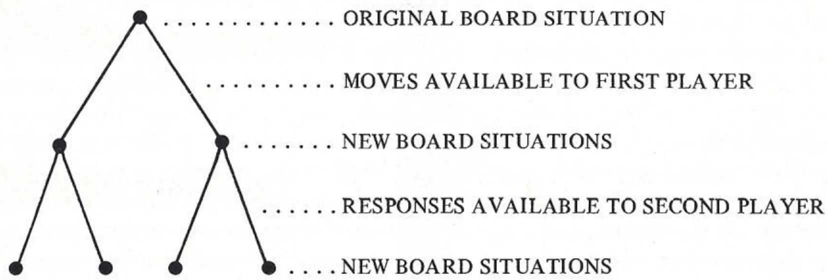


FIGURE 1 Games provide a search environment with a new twist, competition. As shown, the nodes represent game situations and the links represent the moves that connect them.

maximizing player, he is looking for a path leading to a large positive number, and he will assume that his opponent will try to force the play toward situations with strongly negative static evaluations.

Thus in the stylized miniature game tree shown in figure 2, the maximizer might hope to get to the situation yielding a static score of 8. But he knows that the minimizer would not permit that since he can choose a move deflecting the play toward the situation with a score of 1. In general the decision of the maximizer must take cognizance of the attitude of the minimizer at the next level down. If the look ahead goes further, then certainly the minimizer moves in accord with choices of the maximizer at the next level down. This continues until the limit of exploration is reached and the static evaluator provides a direct basis for selecting among alternatives. In the example, the static evaluations at the bottom determine that the choices available to the minimizing player yield expected scores of 2 and 1 at the level just up from the static evaluations. Knowing these scores are expected when the minimizer makes his logical moves, the maximizer can determine the best play for the computer at the next level up. Clearly he moves toward the node from which the minimizer can do no better than to hold the expected score to 2. Again, the scores at one level determine the action and the summary score at the next level up.

■ The process by which the scoring information passes up the computer's game tree is called the minimax process since the scores at a node are either the minimum or maximum of the scores at the node immediately below.

Note that this can be expensive for two reasons: first, calculating the static value for all possible paths may be expensive; and second, even generating all possible paths can be a stumbling block. Which costs more depends on the details of the static evaluator and move generator used.

It might seem at first that the static evaluator must be used on each situation found at the bottom of the tree. But thankfully this is not so. There is a procedure that reduces both the amount of tree that must be generated and the number of static evaluations, thus cutting down on the work to be done overall. It is somewhat like the branch-and-bound idea in that some paths are demonstrably nonoptimal even though not followed to the limit of look ahead.

Consider the situation in the first drawing of figure 3 in which the static evaluator has already been used on the

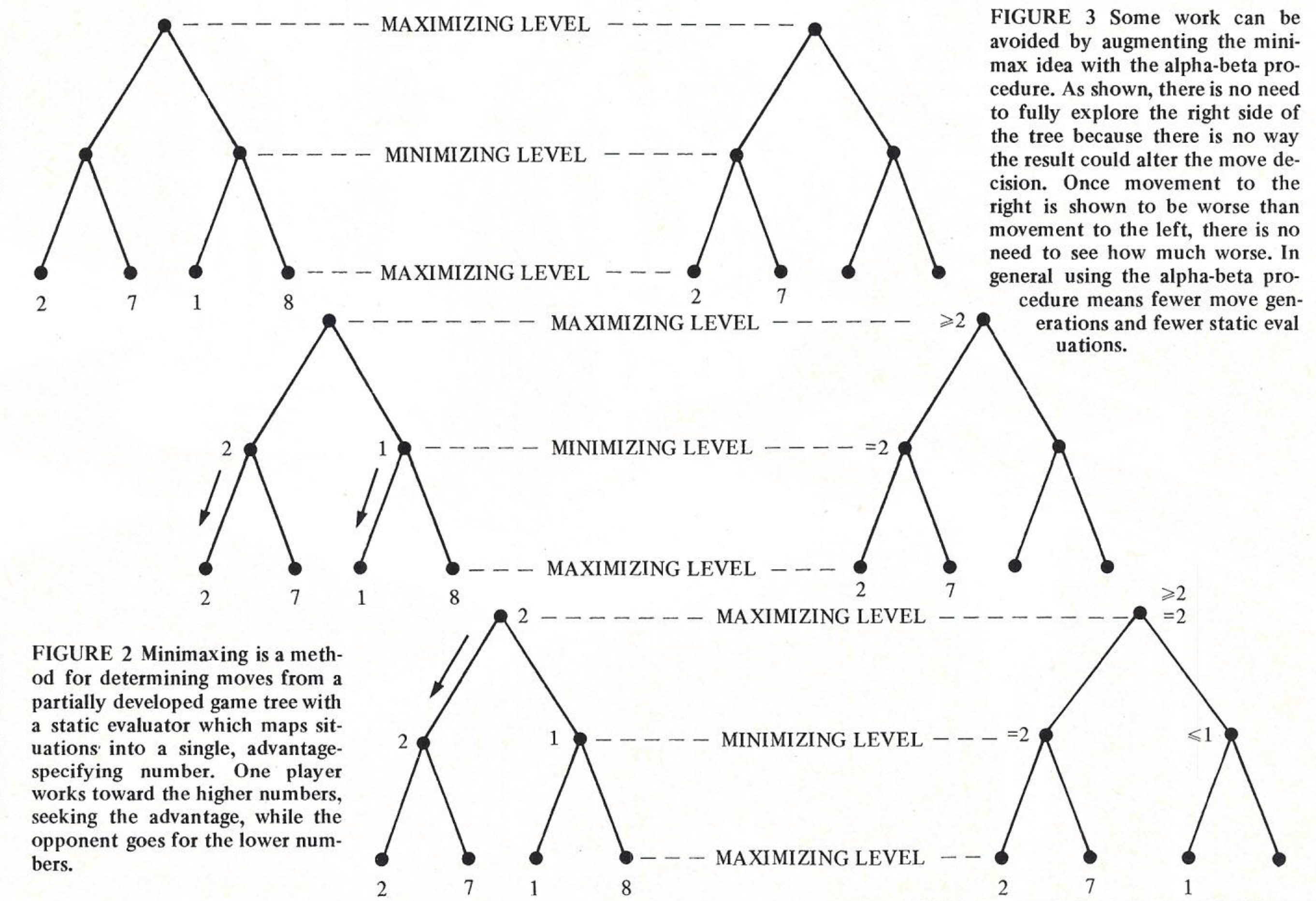


FIGURE 3 Some work can be avoided by augmenting the minimax idea with the alpha-beta procedure. As shown, there is no need to fully explore the right side of the tree because there is no way the result could alter the move decision. Once movement to the right is shown to be worse than movement to the left, there is no need to see how much worse. In general using the alpha-beta procedure means fewer move generations and fewer static evaluations.

FIGURE 2 Minimaxing is a method for determining moves from a partially developed game tree with a static evaluator which maps situations into a single, advantage-specifying number. One player works toward the higher numbers, seeking the advantage, while the opponent goes for the lower numbers.

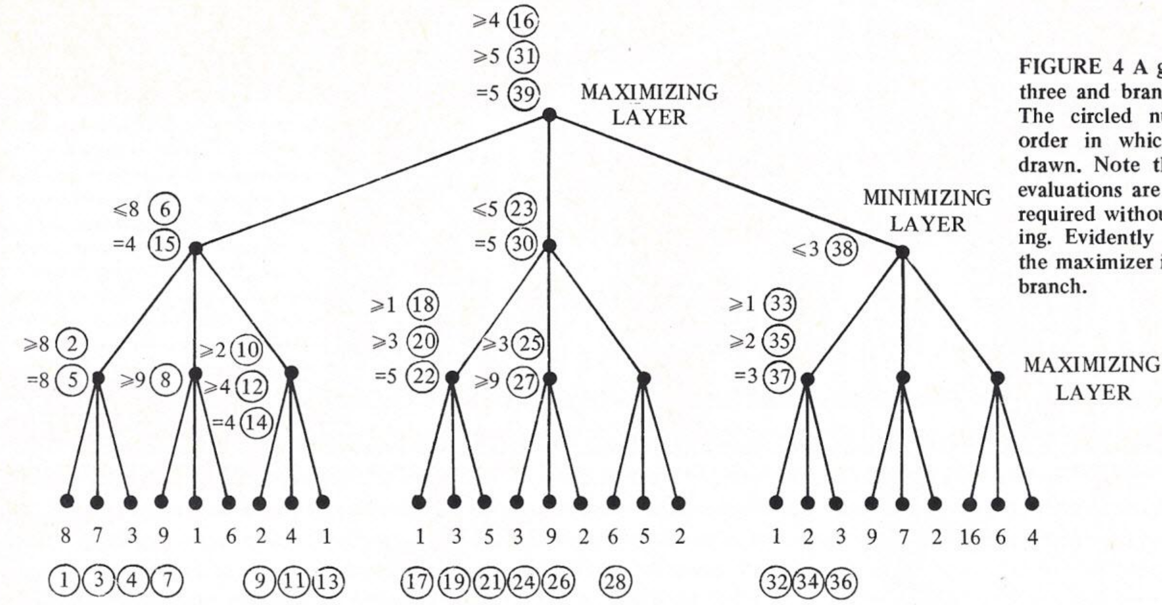


FIGURE 4 A game tree of depth three and branching factor three. The circled numbers show the order in which conclusions are drawn. Note that only 16 static evaluations are made, not the 27 required without alpha-beta pruning. Evidently the best play for the maximizer is down the middle branch.

first two terminal situations. Minimaxing on the scores of 2 and 7 determines that the minimizing player is guaranteed a situation with a score of 2 if the maximizer takes the left branch at the top node. This in turn ensures that the maximizer can do no worse than direct play toward a score of 2 at the top. This is clear even before any other static evaluations are made since the maximizer can certainly elect the explored branch if others turn out worse. This is indicated at the top node in the second drawing of figure 3.

Now suppose the next node is evaluated and produces a score of 1. Seeing this, surely the minimizer can do no worse than 1 by the same reasoning that showed that the maximizer can do no worse than 2 at the top. Only the sense of *worse* has changed. At the maximizer levels, worse means toward the smaller numbers, while at the minimizer levels, worse means toward the higher.

Look closely at the tree. Does it make sense to go on to the board situation at the final node? Can the value produced there by the static evaluator possibly matter? Strangely the answer is *No*! For surely if the maximizer knows he can do no worse than 2 along the left branch, he will need to know no more about the right-hand branch other than that he can do no better than 1 there. The last node evaluated could be +100 or -100 or any number whatever without affecting the result.

On reflection, it is clear that use has been made of the following key principle:

■ If the opponent has one response that establishes a potential move as bad, there is no need to check any other responses to the potential move. If one is gored, there is certainly no need to find out how many ways or how badly in the worst case.

This principle is the foundation of the *alpha-beta* technique. In practice the actual procedure consists of two activities:

■ Whenever something is discovered about the best that can be hoped for at a given node, check what is known about the parent node. It may be that no further work is sensible below the given node.

■ Whenever the exact game value of a node is established, check what is known about the nodes above. It may be that the best that can be hoped for at the parent node can be revised or determined exactly.

It is appropriate now to see how the computer's alpha-beta technique applies to a larger example. Unfortunately, it is a bit difficult to see how static evaluations intermix with conclusions about node values on paper. A lecture or movie would be better, but lacking that, we make do with circled even numbers placed beside each conclusion showing the order in which they were determined. These are shown in the example of figure 4 in which we look at another stylized tree for which three branches descend from every node.

■ *1-2*: Moving down the left branch at every decision point, the search penetrates to the bottom where a static value of 8 is unearthed. This 8 clearly means that the maximizer can do no worse than 8 with the three voices available. A note to this effect is placed by step 2.

■ *3-5*: To be sure nothing better than 8 can be found, the maximizer examines the two other moves available to him. Since 7 and 3 both indicate inferior moves, he concludes that the score achievable is exactly 8 and the correct move is the first one examined.

■ *6*: Nailing down the maximizer's score at the lowest node enables a conclusion about what the minimizer can hope for at the next level up. Since one move is now known to lead to a situation that gives the maximizer a score of 8, then the minimizer can do no worse than 8 here.

■ *7-8*: To see if the minimizer can do better at the second level, his two remaining moves must be examined. The first leads to a situation from which the maximizer can score at least a 9. Here cutoff occurs. By taking the left branch, the minimizer forces a score of 8, but by taking the middle branch, the minimizer will do no better than 9 and could do worse if the other maximizer choices are even more positive. Hence the middle branch is bad for the minimizer, there is no need to go on to find out how much worse it is, and there is consequently no need for two static evaluations. There is no change in the minimizer's worst case expectation – it is still 8.

■ *9-14*: The minimizer must still investigate his last option, the one to the right. This in turn means seeing what the maximizer can do there. The next series of steps bounces between static evaluations and conclusions about the maximizer's situation im-

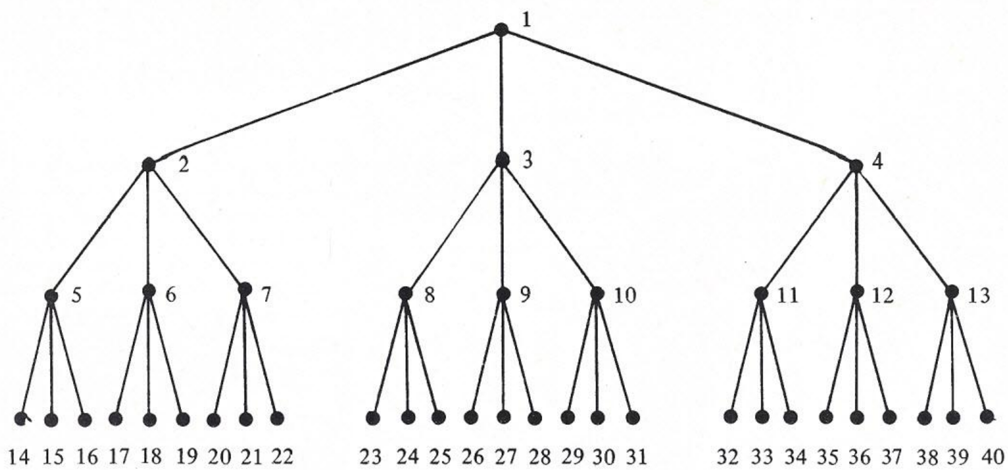


FIGURE 5 In a perfectly ordered tree, the alpha-beta procedure cuts the exponent of combinatorial explosion in half. This is because all of the adversary's options need not be considered in verifying the left-branch choices. Perfect ordering of a tree with depth three and branching factor three reduces the number of required static evaluations from 27 to 11.

mediately above them. The conclusion is that the maximizer's score is 4.

■ *15*: Discovering that the right branch leads to a forced score of 4, the minimizer takes the right branch since 4 compares favorably with 8, the previously low score.

■ *16*: Now a bound can be placed at the top level. The maximizer, surveying the situation there, sees that his left branch leads to a score of 4. He now knows he will get at least that. To see if he can do better, he must look at his middle and right branches.

■ *17-22*: Deciding how the minimizer will react at the end of the middle branch requires knowing what happens along the left branch descending from there. Here the maximizer is in action discovering that the best play is to a position with a score of 5.

■ *23*: Until something definite was known about what the maximizer could could do, no bounds could be placed on the minimizer's potential. Knowing that the maximizer gets 5 along the left branch, however, is knowing something definite. The conclusion is that the minimizer can do no worse than 5.

■ *24-27*: In working out what the maximizer can do below the minimizer's middle branch, it is discovered part way in the analysis that the maximizer can reach a score of 9. But 9 is poor relative to the known fact that the minimizer has one option that insures a 5. Cutoff occurs again. There is no point in investigating other maximizer options, thus avoiding one static evaluation.

■ *28-29*: Looking at the minimizer's right branch quickly shows that it too gives the maximizer a chance to force the play to a worse score than the minimizer can achieve along the left branch. Cutoff saves two static evaluation here.

■ *30*: Since there are no more branches to investigate, the bound of 5 on the minimizer's score becomes not a bound, but the actual value achievable.

■ *31*: The maximizer at the top, seeing a better deal through the middle branch, chooses it tentatively and knows now that he can do no worse than 5.

■ *32-37*: Now the maximizer's right-branch choice at the top must be explored. Diving into the tree, bouncing about a bit, leads to the conclusion that the minimizer sees a left branch choice insuring a score of 3.

■ *38*: The minimizer can conclude that the left branch score is a bound on how well he can do.

■ *39*: Knowing the minimizer can force play to a situation with a score of 3, the maximizer at the top level concludes there is no point in exploring the right branch further. After all, a score of 5 follows a middle branch move. Note that this saves not only six static evaluations but also two applications of the move generator which may be a considerable saving in itself.

It is not unusual to get lost in this demonstration. Even seasoned game specialists feel magic in the alpha-beta technique each time they ponder it after a long interval. Each individual conclusion seems right, but somehow the global result is strange.

Note, incidentally, that in the example it was never necessary to look more than one level up in order to decide whether or not to stop exploration. This was strictly a consequence of the shallowness of the depth-three tree used. With trees of depth four or more, so-called deep cutoffs can occur which require looking further.

It is important to understand just what this alpha-beta technique can be expected to do. One way of addressing this is to ask about the best and worst cases. In the worst case, alpha-beta does nothing. It is easy to construct a tree in such a way as to ensure that the static evaluator must be applied to all terminal situations.

Analyzing what can come of the best case requires some work. Suppose the tree, by luck or otherwise, is ordered with each player's best move being the left-most alternative at every node. Then clearly the best move of the player at the top is to the left. But how many static evaluations are needed for the top-most player to be sure that this move is optimum? To work into the question, consider the tree of depth 3 and branching factor 3 shown in figure 5.

Presuming that the best moves for both players are always to the left, then the value of the left-most move for the maximizing player at the top is the static evaluation found for the board situation at the extreme bottom left. Assuming this is correct, then the maximizer has something concrete against which the quality of the alternatives can be compared. He need not consider all of his opponent's replies to those alternatives, however.

■ To verify the correct move at a given node in an ordered tree, it is necessary to consider relatively few of the terminals descendent from the immediate alternatives to the move to be verified. This is true because all terminals found below nonoptimal moves by the opponent can be ignored.

Why is it necessary to deal with all the options of the moving player while ignoring all but one of his opponent's moves? This is a sticky point. The explanation requires close attention to the basic idea: if the opponent has some series of responses that make a move bad no matter what series of moves the moving player might choose, then the move must be bad.

The key to understanding lies in the words *some,* and *no matter what.* The *some* suggests trying one of the op-

ponent's moves wherever he has a choice and hoping that it is good enough to certify the conclusion. But to be sure that the conclusion holds no matter what the moving player might do, it is clearly necessary to check out all of his alternatives wherever he has a choice.

Thus the hope in the example is that only the left-most move from node 3 to node 8 will need exploration. All of the maximizer's counter-responses to that move must be checked, so static evaluations need to be made at nodes 23, 24, and 25.

These establish the maximizer's score at 8, which in turn sets an upper bound on what the minimizer can do at 3, which finally by comparison with the minimizer's score at 2 should show that no further work below node 3 makes any sense. Similar logic applies to node 4 leading to static evaluations at 32, 33, and 34.

But now how can the maximizer be sure that the score transferred up the left edge is valid? Surely he must verify that an intelligent minimizer at 2 would select the left-most branch. This verification can be done by assuming the number coming up the left edge from 5 is correct and then rejecting the alternatives as efficiently as possible. But by the same arguments used at 1, it is clear that not all of the minimizer's opponent's options need be examined. Again branching occurs only at every second level working out from the choice to be verified along the left edge. Static evaluations must be done at 17 and 20.

Finally there is the question of the minimizer's assumption about the number coming up from 5. This requires exploring all of the maximizer's alternatives, the trivial case of the general argument, resulting in static evaluations at 15 and 16 in order to be sure that the static evaluation done at 14 yields the right number to transfer up.

Thus only 11 of the 27 possible static evaluations need be made in order to discover the best move by the computer when by luck the alternatives in the tree have been nicely ordered. In deeper trees, with more branching, the saving is more dramatic. In fact it can be demonstrated that the number of static evaluations needed to discover the best move in an optimally arranged tree is given by

<number of evaluations>
$= 2b^{d/2} - 1$ for d even
$= b^{(d+1)/2} + b^{(d-1)/2} - 1$ for d odd
where b = branching factor
and d = depth of search in moves.

A straighforwarded proof by induction verifies the formula. One need only generalize the line of argument used in the last example, focusing on the idea that verification of a choice requires full investigation only every second level. Note that the formula is certainly correct for $d = 1$ since it then simplifies to b. For $d = 3$ and $b = 3$, the formula yields an answer of 11, which nicely verifies the conclusion reached for the example.

Warning: the formula is only for the special case in which the tree is perfectly arranged. As such it is an unrealistic estimate of what can actually be expected, for if there were a way of arranging the tree with the best moves on the left, clearly there would be no point in using alpha-beta pruning at all! But this is not to say that the exercise has been fruitless. It establishes the lower bound on the number of static evaluations that would be needed in a real game. It is a lower bound that may or may not be close to the real result depending on how well the moves are in fact arranged. The real result must lie somewhere between the worst case, at which all b^d terminals must be evaluated, and the best case, which still requires approximately $2b^{d/2}$ terminal evaluations.

Either way the number required becomes impossibly large with increasing depth. Alpha-beta merely wins a temporary reprieve from the impact of the explosive, exponential growth. It does not prevent it. See figure 6.

The alpha-beta search speedup technique guarantees as good an answer as can be found by complete, exhaustive minimaxing. Many other techniques have no such guarantee but are used in combination with alpha-beta pruning as additional weapons against explosive tree growth. The following are typical:

Limiting breadth. A brute-force way of reducing the effective branching factor in a game tree is to ignore the less likely possibilities. Since a plausible move generator is generally used anyway in connection with the alpha-beta method, it is easy to single out only the most likely descendants from any node for study.

To generalize slightly, one may arrange for breadth to vary with depth of penetration, possibly even biasing efforts toward moves of greater plausibility through some formula like the following:

<number of descendants>
= <number of descendants from parent>
– <rank in plausibility>

Thus if a node is one of five descendants and ranks second most plausible

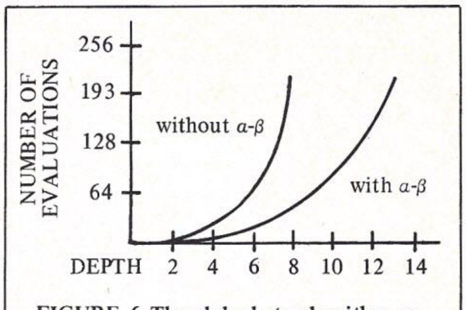


FIGURE 6 The alpha-beta algorithm reduces the rate of combinatorial explosion, but does not prevent it.

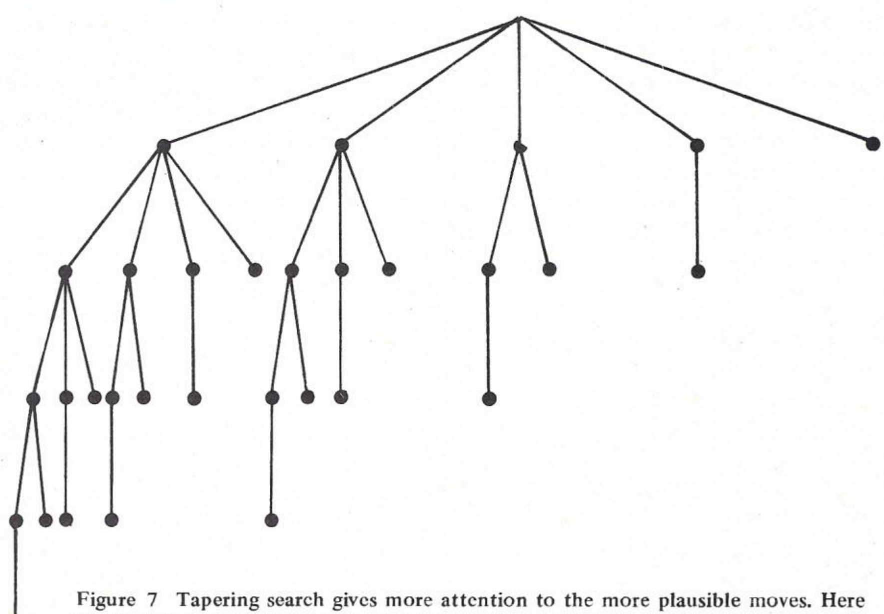


Figure 7 Tapering search gives more attention to the more plausible moves. Here the tapering algorithm reduces the search with increases in depth and decreases in plausibility.

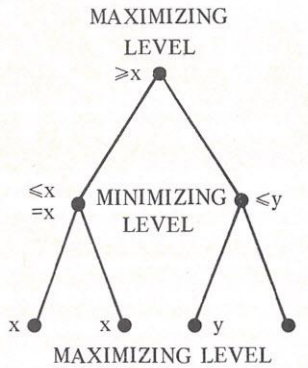


FIGURE 8 Sometimes it makes no sense to explore a branch even though it may be optimal because the potential gain is slight. Here search has shown that the maximizer gets an assured score of x by taking the left branch. Partial analysis of the right branch shows that the maximizer cannot do better than a score of y there. If y is greater than x, there is no alpha-beta cutoff, but if the difference is small, the maximizer may not wish to insist on full evaluation, taking the comfortable left-branch route instead.

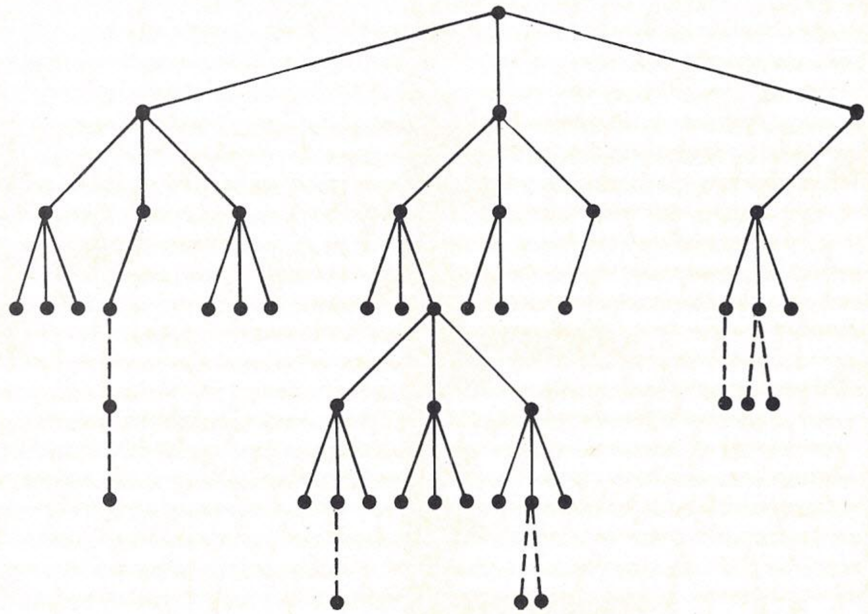


FIGURE 9 Secondary search and feedover procedures may extend exploration beyond normal limits. Dotted lines are continuations of paths which ordinary search had terminated at particular dynamic situations such as imminent capture. The small search dangling beneath the main tree is done to check the static evaluation of the winning streak.

among those five, then there should be 5 – 2 = 3 descendants. A complete tree formed using this idea takes on a tapered look as in figure 7.

Needless to say a tactical breadth limiting heuristic acts in opposition to lines of play that temporarily forfeit pieces for eventual position advantage. Because they trim off the moves that appear bad on the surface, breadth-limiting programs are unlikely to discover spectacular moves that seem disastrous for a time, but then win back everything lost and more.

Disaster cutoff. Another way of limiting computer search down through bad moves depends not on the simple position of a move in the plausibility ranking but rather on some accumulating plausibility value. This makes the shape and size of the tree sensitive to the particular situation. If only one line of play makes any sense at all, it would be the only one pursued, and could therefore be followed to greater depth. On the other hand, if many lines of play seem equally plausible, then this method tends to allocate resources among them evenly.

Futility cutoff. Pushing the tree down equally through moves of equal plausibility makes sense. But after some solid evaluations are made, it may well develop that considerable search saving can be had by rejecting partially explored moves that can at best offer only slight improvement on some fully explored move.

Figure 8 illustrates the point. The tree as shown has already been partially analyzed. It is convenient to use *x* and *y* for the static evaluations rather than sample numbers. Note that the maximizing player at the top has a sure score of *x* or higher.

Now assume *y* is bigger than *x* but only by a little. This is a key point to keep in mind. Down the right branch from the top, the minimizer can hold the maximizer to a score of *y* or lower – lower if and only if the yet unevaluated node in fact produces a score lower than *y*.

Now as long as *y* is greater than *x* there is no ordinary alpha-beta cutoff because the minimizer cannot demonstrably force the play to a low enough score. But, if the difference between *x* and *y* is small, no matter how good the unevaluated node might be, the potential gain achievable by completing the computer search is small. This is true because if the unevaluated node is very good, the minimizer will simply avoid it, holding the maximizer to a score of *y*, a meager gain of $y - x$ over what he knew he could get by going down the left side. In this case the decision might well be to move to the left, accepting the certain score of *x*, and avoiding the extra static evaluation. To be sure, if the value of the unexplored node is bigger than *x*, then complete analysis would show the absolutely optimal path to be to the right. But why should a program work like mad for little potential gain? The score found is likely to be lower than *x* anyway, making the additional work completely futile.

Feedover conditions. So far the techniques limit tree growth. Sometimes it is useful to go the other way and extend the tree when circumstances warrant. If, for example, a chess configuration is particularly dynamic, it makes sense to continue the play until a more quiescent state obtains. There are a number of so-called feedover conditions that can cause a configuration to be considered dynamic and deserving of further exploration by the computer:

The king is in danger.

A piece exchange is imminent.

A pawn is about to become a queen.

Secondary search. After all ordinary search is complete with the alpha-beta method, the various heuristic methods and feedover all playing their assigned parts, it is often good to grow a small secondary tree down from the node judged to be at the bottom of the optimal path. Doing this double checks the accuracy of the value assumed for the target node: one estimate comes straight from the static evaluator, and the other from minimax analysis of the secondary search tree. The hope is that the two scores will be rough-

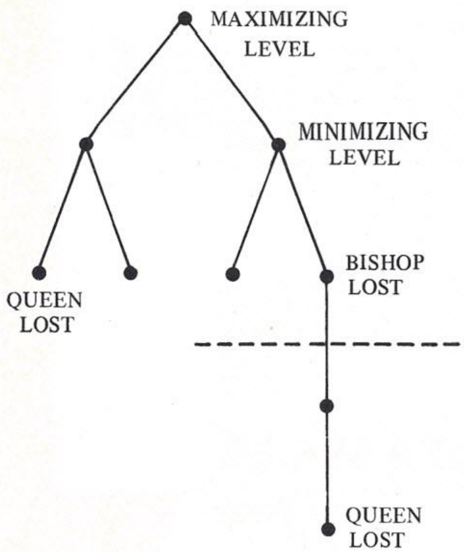


FIGURE 10 The horizon effect foils search-oriented game procedures when disasters can be delayed but not prevented. Here the left-branch move leads directly to queen loss, and the right-branch move leads to a sacrifice followed by queen loss.

ly the same, confirming that no surprises lie just beyond the limit encompassed by primary search. Presumably the extra depth of search is impractical for the tree in general, but allowable in the singular case of confirming the estimated value of the optimal line of play. Figure 9 shows the shape a tree may take on when feedover and secondary search are combined.

The surprises spoken of, incidentally, are often the result of delaying moves that put off, but do not prevent disasters. To illustrate how this can work, suppose there is a combination which leads inevitably to queen capture. Suppose further that the side losing the queen can delay the loss by sacrificing other, lesser pieces somewhere else on the board. As long as the queen loss is inevitable, it is absurd to use delaying moves that sacrifice still more material. But these delaying moves may carry the play beyond the depth limits established for the computer tree search. In the sample of figure 10, the apparent choice is between losing the queen or losing a bishop. But the real choice is between losing a queen or losing the queen *and* bishop. Metaphorically speaking, an inevitable loss is pushed beyond the field of view by a lesser sacrifice. The phenomenon commonly is called the horizon effect. Secondary search helps somewhat, but does not yield a total solution.

Summary

- The simplest search techniques are depth-first and breadth-first search. Neither finds optimal paths, and both may be inefficient.
- Hill climbing is an improvement over depth-first search because the branches under any node are explored in the order of their plausibility. Hill climbing still has the same problems inherent in depth-first search, however, in that a wrong decision high up in the tree can lead to endless thrashing low down.
- Context-sensitive, best-first searches push forward from the most promising node yet encountered.
- The branch-and-bound technique is a fundamental method for finding optimum routes. The basic idea is to extend the developing tree from the end of the least costly partial path. This is often improved through the use of underestimates of the distances remaining to the goal.
- Searching for solutions to network constraints is best done by a range-constriction approach that carries each alternative forward until it is definitely out of the running. The method seems applicable in understanding language as well as in understanding scenes.
- The adversary nature of games makes game tree search a very special subject. The minimax idea is the foundation on which all else rests. Minimax is the procedure by which conclusions about what to do at the lowest nodes of the search tree percolate up to determine what happens at the top.
- The alpha-beta technique depends on a fact that once the opponent has one way to insure that a move is disastrous, there is no need to explore that move further. While alpha-beta pruning saves considerable computer work, the optimum line of play is found nonetheless.
- Alpha-beta pruning may be augmented by a number of heuristic pruning techniques. All of these introduce some danger that the optimum play may not be selected.

References

- Nils J. Nilsson, *Problem Solving Methods in Artificial Intelligence,* McGraw-Hill Book Company, New York, 1971. A thorough textbook treatment of search.
- David Waltz, "Understanding Line Drawings of Scenes with Shadows," in *The Psychology of Computer Vision,* edited by Patrick Henry Winston, McGraw-Hill Book Company, New York, 1975. Gives a detailed description of the range-constriction search algorithm Waltz uses in scene analysis.
- Gerald Jay Sussman and Richard Matthew Stallman, "Heuristic Techniques in Computer-Aided Circuit Analysis," AI-M-328, The Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1975. Demonstrates that Waltz-like search is useful in expert problem solving by showing that common-sense circuit analysis can be done using constraint propagation.
- Claude E. Shannon, "Automatic Chess Player," *Scientific American,* vol. 182, no. 48, 1950, and "Programming a Digital Computer for Playing Chess," *Philosophy Magazine,* vol. 41, March, 1950. Classic papers on chess which anticipated most of the subsequent work to date.
- Allen Newell and Herbert A. Simon, *Human Problem Solving,* Prentice-Hall, Englewood Cliffs, New Jersey, 1972. A thick book containing a long chapter describing many chess ideas developed by the authors since they began thinking about chess in the middle 1950s.
- Arnold K. Griffith, "A Comparison and Evaluation of Three Machine Learning Procedures as Applied to the Game of Checkers," *Artificial Intelligence,* vol. 5, no. 2, Summer 1974. The idea of search and fancy static evaluation is artfully contested in this paper.
- James J. Gillogly, "The Technology Chess Program," *Artificial Intelligence,* vol. 3, no. 3, 1972. A good paper showing what can be done by brute force search.
- Hans Berliner, "Chess as Problem Solving: The Development of a Tactics Analyzer," PhD thesis, Department of Computer Science, Carnegie-Mellon University, Pittsburgh, Pennsylvania, 1975. Deals extensively with chess. Berliner advocates sophisticated, goal-directed plausible move generation, drastically trimmed search trees, and dynamically determined search depth.

The next two papers probe deeply into the alpha-beta idea with a view toward understanding expected performance rather than best or worst case performance. It is shown that the problem is not easy.

- S.H. Fuller, J.G. Gaschnig, and J.J. Gillogly, "Analysis of the Alpha-Beta Pruning Algorithm," Department of Computer Science, Carnegie-Mellon University, Pittsburgh, Pennsylvania, 1973.
- Donald E. Knuth and Ronald W. Moore, "An Analysis of Alpha-Beta Pruning," *Artificial Intelligence,* vol. 6, No. 4, 1975.

The next two papers by Samuel are classics which deal extensively with tree-pruning heuristics, schemes for combining evidence, and methods for adpative parameter improvement.

- A.L. Samuel, "Some Studies In Machine Learning Using the Game of Checkers," *IBM Journal of Research and Development,* vol. 3, no. 3, 1959.
- A.L. Samuel, "Some Studies in Machine Learning Using the Game of Checkers II. Recent Progress," *IBM Journal of Research and Development,* vol. 11, no. 6, 1967.

. . . Try, Try Again

BY ROBERT L. GLASS

Tales of Computing Folk: Hot Dogs & Mixed Nuts *chronicles the era of the free-thinking, free-swinging computing iconoclast, now banished from the halls of industry to the microcomputer in his garage. It both mourns and celebrates the replacement of the industrial software craftsman with a team-oriented, standards-constrained, assembly-line producer of structured software. The stories the book tells are true; the names have been changed to protect the innocent – and the guilty. One of these stories – really Part Two of a two-part story – appears below. Part One, "A Computer Project Which Failed", was published in the April* Personal Computing. *It told the story of a computer hobbyist who converted his ideas into money, only to see the money evaporate as his concepts crashed around him. Part Two bears the happy message that failure is not irrevocable.*

How do you take a computing project which failed and convert it into a head-line making business success?

Rethink your marketing strategy. At least, that was the answer Joe Bill Jackson found after the demise of his Data Touch computer system.

You may remember the story of Data Touch. Joe Bill sold Ring-a-ding Tel and Tel on the idea of computer power in the home. The touch-tone phone was the vehicle for his call-up service. Even Ring-a-ding President Ma Bile came to their opening of a pilot version of the service in Scenic, Kansas.

But Joe Bill's pride had turned to despair as the customers stayed away in droves. And when he finally unplugged Data Touch and left Scenic, it was as an economic failure.

Philosophers say that failure is the severest test of a person's inner strengths. Well, Joe Bill was mightily tested. When his dream died, a little of the dreamer died with it. But after a mourning period, Joe Bill started to struggle back.

"What really went wrong?" was the question that kept reverberating through Joe Bill's brain. His common sense, his business knowledge, and even a professional marketing survey had all declared the idea a sure-fire winner.

As the months ticked by in Joe Bill's psychological exile, he struggled to grasp the shape and substance of the failure. He decided to remold Data Touch and try again. But he had to understand the reasons for failure in order to avoid them next time.

His first major problem, Joe Bill realized, was to quit thinking as a technologist. He had a product to market to the public. The public doesn't buy technology, he now knew. The public buys service.

A small glimmer of light formed at the end of his mental tunnel. He grabbed a plane, flew back to Scenic, and sifted through the archived data on Data Touch usage. The picture wasn't clear, but there were trends.

The income tax calculator service had been a solid but not outstanding performer. The desk calculator function had been a total loser – the cheapie hand-held calculator wiped him out there. Data storage had been little-used – apparently the public really didn't know what data it cared to store and retrieve. But the automated bill-paying service? An ah-ha began to form in Joe Bill's brain. The bill-paying service had been a roaring success. Only a shortage of merchants willing to accept the Data Touch method of payment had prevented bill-paying from saving the Data Touch day.

Joe Bill narrowed his thinking to the bill-paying problem, stripping away in his mind as impedimenta the unsuccessful services. It appeared that the secret of success was the attraction of merchants. But how could he do that? And if he was not up to that task, who could best do it for him?

Banks. The obvious answer was banks. And suddenly he was offering a service which was not a technology toy – it was a bill-paying service. No, it wasn't a bill-paying service. It was (and a ta-da exclamation mark leapt into his brain) Electronic Funds Transfer. EFT, everybody's promise and fear, was the service he was better prepared to offer than anyone else in the country. The implications of his ah-ha were staggering! He had been sitting at the forefront of an explosive new technology, and hadn't even known he was there!

But now he knew. Visions of a new company sprang into his mind. EFT, Inc. Joe Bill began to realize that his future no longer was wedded to the phone company. He could go it alone – with some financial help.

And that brought him right back to banks. Joe Bill laid out a whirlwind tour of major banks across the land, and a pitch to make to them. "Buy into EFT" was his message. "Pump some money into my company and I'll make EFT (nee billing-paying by phone) available to you as a turnkey package."

Well, it worked. Joe Bill lined up a consortium of a half-dozen banks, from Audible Chasm, Nevada, to Open Spaces, Mass., to back him. And another dozen accounts were ready to put out enough money to offer the service. And suddenly, yesterday's failure became today's success and tomorrow's potential millionaire. It was almost that quick.

The moral of the story? It's too traditional, too predictable to even say. But it starts out with "If at first you don't succeed . . . ." □

Reprinted by permission. Robert L. Glass, Tales of Computing Folk: Hot Dogs and Mixed Nuts. *Computing Trends, 3325 Lakewood Ave. S., Seattle, WA 98144. ©1978. 100 pages, $8.50 (0.45 postage/handling).*

Investment Analysis

BY
WILLIAM LAPPEN

If you were offered a choice between receiving $120 ten years from now or $10 each year for 10 years, which choice would you make? If these choices were being *sold* to you, how much would you be willing to pay? To answer these questions, one must understand the concept of "present value". This article develops the basis of "present value" then applies it to investments in general and apartment house investments in specific.

Putting $100 in a bank account that pays 6% annual interest, you would receive $106 at the end of the first year – your original $100 plus $6 interest. At the end of the second year, the amount would have grown to $112.36 ($106 * 1.06). The formula for this "compounding" of interest is:

$$FV = PV * (1 + R)^N$$

FV is the value in the future (future value)
PV is the value today (present value)
R is the applicable interest rate per period
N is the number of periods.

Using this formula, we can determine the value today of $120 received in 10 years: $PV = FV/(1+R)^N$. If we use 6 percent as a rate, the present value will be $67.01. That means that if you invest $67.01 in a bank account today and earn 6% interest per year on that account, the account balance will be $120 at the end of 10 years.

Instead of getting a "lump sum" in the future, you prefer to get $10 each year, then the first $10 will be worth only $9.43 because you will have to wait one year for it. (Remember, if you invested $9.43 today, you would have $10 at the end of one year.) The next $10 will be worth $8.90 today. The present value of a "stream" of income, therefore, is equal to the sum of the present values of the money received. The formula is:

$$PV = PMT*((1-(1+R)^{-N})/R)$$

PV is the present value
PMT is the payment per year
R is the periodic interest rate
N is the number of periods.

In the example of $10 per year, the present value would be $10 * ((1-(1+0.06)^{-10})/0.06) = 73.60$.

Going back to the original example, you would choose $10 per year for ten years over $120 in 10 years if you could earn 6% on your funds. You would be willing to pay up to $73.60 for your preferred choice and up to $67.01 for $120 in ten years.

The above example ignores taxes (a procedure not highly recommended). To reflect income taxes, assume that you are presented with the opportunity to "buy" $100 per year for 4 years and a bank is willing to lend you $200 to help purchase this investment. How much should you pay for this investment if you are in the 40% tax bracket and the bank loan is for four years at 7%? First, it is necessary to see how much of that $100 you will get to keep each year. The bank and Uncle Sam are both taking some of it away. But, the money you pay to the bank for interest is later deductible on your income taxes as an expense. To fully "amortize" (pay) the bank loan, go back to the second formula and solve for PMT, $59.05.

The interest for the first year is $14 ($200 * .07) and the remainder of the $59.05 ($45.05) is used to reduce the amount you owe the bank. In the second year, the interest is 7% of the outstanding principal ($200 - $45.05) or $10.85. The principal is then reduced by $48.20 to $106.75 . . .

TABLE I

	Year 1	Year 2	Year 3	Year 4
Gross Income	100.00	100.00	100.00	100.00
Interest	-14.00	-10.85	-7.47	-3.86
Taxable Income	86.00	89.15	92.53	96.14
Tax (40%)	34.40	35.66	37.01	38.46
Gross Income	100.00	100.00	100.00	100.00
Loan Payment	-59.05	-59.05	-59.05	-59.05
Tax Payments	-34.40	-35.66	-37.01	-38.46
CASH FLOW	6.55	5.29	3.94	2.49

Taking the present values of "Cash Flows," you can determine the amount you would be willing to pay for such an investment. If 6% is the rate that you get by putting your money into a savings account, that then is the figure you would use. However, you have to pay taxes on that 6% and the cash flows already reflect taxes. Therefore the proper "discount" rate is 3.6% (6% * (1-40%)), which is the "after tax rate." Using the calculations for present value:

TABLE II

YEAR	CASH FLOW	DISCOUNTED
1	6.55	6.32
2	5.29	4.93
3	3.94	3.54
4	2.49	2.16
		16.95

You would be willing to pay up to $16.95 for the investment. If the price was more, you would be better off putting your money in the bank and earning the 6% (3.6% after taxes).

A quick glance at Table 1 will show you the merit of using a computer to "crank out" these figures. The columnar structure should suggest arrays. Of course, the computer can discount Cash Flows and tell you how much you should pay for the investment.

As long as you are going to put this on your computer, let's complicate the problem! Assume you are presented with the opportunity to invest in an apartment building (become a landlord).

The apartment building figures show a return of $1,000 per year after operating expenses (maintenance and repairs). You can obtain a bank loan for $18,000 for 30 years at 8% per year (compounded monthly). The owner is so eager to sell the property, he offers to lend you $5,000 for 5 years at 10% (compounded monthly). Your tax rate is 40% and the asking price is $30,000. Should you buy the apartment building? How much should you pay?

One advantage of apartment house ownership is tax deductions. The federal government allows you to deduct depreciation from the income of the building. If you figure that the building has about 25 years left before you replace it, then you may deduct, each year, 1/25 of the amount paid for the building (excluding the amount you paid for the land). But Uncle Sam is even nicer, he allows you to take "accelerated depreciation." This form of depreciation is basically like declining balances on loans. In the first year, for example, you may deduct 125% of the "straight line" rate (straight depreciation). Assume that the *building* cost is $15,000. Therefore, the first year's depreciation is $15,000/ 25*1.25 = $750. This amount is deducted from the original price of the building ($15,000) and the remainder ($14,250) is depreciated the same way next year ($712.50). . . .

One difference between apartment-house income and some other investment is that apartment income changes annually. Assume that you expect to raise rents 5% per year and expect to face increasing operating costs of 5%. Therefore, your income after operating expenses will grow at the rate of 5%. Let's analyze this investment for 6 years. Because most of the numbers are estimates, I will round the numbers; but interest on the loans will be computed monthly. Using the following familiar format you will notice that the taxable income is negative:

TABLE III

	YEAR 1	YEAR 2	YEAR 3	YEAR 4	YEAR 5	YEAR 6
Net Income	1050	1100	1160	1220	1280	1340
Interest						
First Loan	-1430	-1420	-1410	-1390	-1380	-1360
Second Loan	-460	-380	-280	-180	-70	0
Depreciation	-750	-710	-680	-640	-610	-580
Taxable Income	-1590	-1410	-1210	-990	-780	-600
Tax Benefits	630	560	480	390	310	240
Net Income	1050	1100	1160	1220	1280	1340
Tax Benefits	630	560	480	390	310	240
Loan Payments						
First	-1580	-1580	-1580	-1580	-1580	-1580
Second	-1270	-1270	-1270	-1270	-1270	0
Cash Flow	-1170	-1190	-1210	-1240	-1260	0

This means you are now able to "shelter" other income, thus saving on taxes. This is like receiving extra income from the building. Note also that there is a negative cash flow for the first 5 years (until the second loan is paid). This means that even after tax benefits from apartment-house ownership, you will still have to spend money each year to "carry" the building.

Even though this doesn't look too good, remember that we are only looking at the first 6 years. Let's assume you decide to sell the apartment house now. In spite of the fact that you took "depreciation," you should still be able to get more than you paid for the building. Generally, prices follow income and your income has gone up 5% per year. Therefore, the original price of $30,000 will be about $40,200 in six years.

After paying off the remaining balance on the first loan ($16,880) and paying a five-percent brokerage commission to sell the building ($2,010), you are still left with a gain of $21,310.

However, before you can bank that profit, you must pay taxes on it. Uncle Sam has to eat too. The gain you made on the building is calculated by taking the final sales price and subtracting selling expenses (brokerage commission) plus the original purchase price as well as adding that straight-line depreciation you could have taken. In this case, the profit would be $40,200 - $2,010 - $30,000 + ($15,000/25*6) = $11,790. Because you have held the building for over one year, you qualify for capital gains treatment. That means you will only be taxed on one-half of this gain ($5,895). Your capital gains tax would therefore amount to $2,360 (at 40%). (The "one half" applies only to the first $50,000 of a gain).

You're not done yet, though. Remember the accelerated depreciation? Well, Uncle Sam is not going to let you keep the difference in depreciation between straight line and accelerated. The difference is $3,970 - ($15,000/25*6) = $370. Your tax , therefore, is $150 (at 40%).

Sales Price	$40,200
Commission (5%)	-2,010
Capital Gains Tax	-2,360
Depreciation Recapture	-150
Repay First Loan	-16,880
Net Sales	$18,800

You are now ready to determine whether the original price of $30,000 was acceptable. Assuming you have another investment that yields 12% before tax, we will then use 7.2% (12% * (1 - 40%)) to discount the cash flows:

TABLE IV

Year	Cash Flow	Discounted
1	-1170	$ -1090
2	-1190	-1040
3	-1210	-980
4	-1240	-940
5	-1260	-890
6	0	0
6 (Sale)	18800	12390
	Total	$ 7450

Original loans on the property were $18,000 and $5,000. Because the price was $30,000, you would have to put up $7,000 for the building. Then, $7,450–$7,000=$450, earned.

If you don't have $7,000 to put into an apartment house right now, how about selling an analysis like this to your local real estate brokers? Such brokers are in great need of quality financial analysis, yet they rarely have the time to make an analysis. This kind of analysis is particularly important when trying to structure an investment. The tremendous value to a broker (and his client) is that you can run the same basic investment with minor changes in financing, price, tax rates, inflation rates . . . A broker might need 10 or 15 different runs to cover likely possibilities that a potential buyer might be interested in. Also, the broker may be able to demonstrate that the buyer is concentrating on the wrong objectives, – for example, a 1% difference in loan interest on the second loan may not mean anything in the over-all return.

Brokers may also like such an analytical service so they can interest some current owners. One of the most difficult aspects of the brokerage business is getting listings. With an analysis like this to offer present owners, a broker would have a good selling-point to convince the owner of the benefits of selling or trading his building.

Major buyers may also be clients for this type of analysis. Large investors, for instance, are constantly deluged with investment proposals. They might like to use such an analysis as a screening process. Only investments that provide the minimum required return would be considered.

There are many other potential clients out there who are good prospects for a cash flow analysis. With just a little imagination, you can find them. (How about helping a company decide whether to purchase a new forklift or simply lease it?) The financial community awaits you. ■

Program Notes

The program will run on Radio Shack's 4K machine using Level I. If it is a little large for your machine, use multi-line statements or drop some of the PRINT statements (after deleting line 10).

As the Radio Shack Level I BASIC allows only a single array ("A"), everything, therefore, is done in this array. If your BASIC requires a DIM statement, use DIM A(45). The first Y (years) of array A stores the net income until line 4160 where it is replaced with Cash Flow. The second Y elements contain different costs. The third Y elements contain information about second mortgage payments.

The 15 pieces of data that must be input to the program are titled in the DATA statements (lines 30-50). After entering the information, line 770-800 asks if the information is correct. If it is type Y or YES.

If not, type the item number that needs to be changed.

The information is input and stored in the array starting at position 25. To get input I, request A(K+I). The information is stored:

A(K+1)	Years (maximum of 8)
A(K+2)	Growth Rate (from 0 to 100 – program changes to percent)
A(K+3)	Net Income (in thousands)
A(K+4)	First Mtg (principal amount – in thousands)
A(K+5)	Term (of the first mortgage)
A(K+6)	Interest (annual interest – from 0 to 100 – program converts to percent)
A(K+7)	Second Mtg
A(K+8)	Term
A(K+9)	Interest
A(K+10)	Dep. Life (remaining depreciable life on the building)
A(K+11)	Amount (to be depreciated – in thousands)
A(K+12)	Factor (declining balance factor – use 125 for used real estate and 200 for a brand new building)
A(K+13)	Tax Rate (converted to percent)
A(K+14)	Price (for the whole project – in thousands)
A(K+15)	Return (after-tax return on comparable investments – program converts to percent)

To enter before-tax return and have the program calculate the after-tax equivalent, change line 8100 to E = 1+A(K+15)* (1 - A(K+13)/100)/100.

Consider projecting for more than 8 years and printing out every other year. To make this change, increase K in line 80 and delete line 220 (or change the upper bound from 8). When ready to print, test for an odd number and use 10+(INT (I/2)+1)*6 for the tab position. (For most apartment deals that I have seen, 8 years is quite adequate.)

Note – While there are some users who still believe that "If the computer said it, it must be right!", I have gone to great lengths to make sure that the numbers are rounded. This calls attention to the fact that this program is trying to project future events and there will be some error. If the rounding becomes too large (as a percent of the numbers), convert the program to take numbers in hundreds. To do this, change line 6850 so that the "12" becomes "120" and the "12.5" becomes "125." Change line 110 to B$=CASH FLOWS (00). These should be the only changes needed.

SAMPLE RUN

The outputs appear on the video in the order shown. First, the program prints out CASH FLOWS (000). *After you input the requested information,* NET INCOME *is displayed. Finally,* FINAL SALES INFORMATION.

```
CASH FLOWS (000)

1 YEARS           6
2 GROWTH RATE     7
3 NET INCOME      20
4 FIRST MTG       300
5 TERM            30
6 INTEREST        10
7 SECOND MTG      100
8 TERM            4
9 INTEREST        11
10 DEP. LIFE      25
11 AMOUNT         400
12 FACTOR         125
13 TAX RATE       40
14 PRICE          550
15 RETURN         8         CORRECT (Y/XX)?_
```

```
                         NET INCOME
                  1     2     3     4     5     6
NET INCOME       21    23    25    26    28    30
INTEREST
  FIRST MTG      30    30    30    29    29    29
  SECOND MTG     10     8     5     2     0     0
DEPRECIATION     20    19    18    17    16    15
TAXABLE         -39   -34   -28   -22   -17   -14
TAX BENEFIT      15    13    11     8     6     5

NET INCOME       21    23    25    26    28    30
TAX BENEFITS     15    13    11     8     6     5
MTG PAYMENTS
  FIRST         -32   -32   -32   -32   -32   -32
  SECOND        -31   -31   -31   -31     0     0
CASH FLOW       -27   -27   -27   -29     2     3
                   'ENTER' WHEN READY ?_
```

```
FINAL SALES INFORMATION

SALES PRICE                  825
OUTSTANDING PRINCIPAL        286
COMMISSION (5%)               41
EQUITY                       498
DOWN PAYMENT                 150
CAPITAL GAINS TAX            125
DEPRECIATION RECAP.            4

NET PRESENT VALUE -5

                 'ENTER' WHEN DONE ?_
```

Program Listing

```
10 REM DISCOUNTED CASH FLOWS--APARTMENT HOUSE
30 DATA YEARS, GROWTH RATE, NET INCOME, FIRST MTG, TERM
40 DATA INTEREST, SECOND MTG, TERM, INTEREST, DEP. LIFE
50 DATA AMOUNT, FACTOR, TAX RATE, PRICE, RETURN
80 K=24
100 CLS
110 B$=CASH FLOWS (000)
120 PRINT AT 15; B$
140 FOR I = 1 TO 15
160 READ A$
180 PRINT I; A$; " ";
200 INPUT A(K+I)
220 IF (I=1)*(A(K+I)>8) A(K+I)=8
260 NEXT I
300 GOTO 600
400 RESTORE
440 FOR I = 1 TO X
460 READ A$
470 NEXT I
480 PRINT A$;
490 INPUT A(K+X)
600 RESTORE
620 CLS
640 PRINT AT 20; B$;
650 FOR I = 1 TO 15
700 READ A$
720 PRINT: PRINT I; A$, A(K+I);
750 NEXT I
770 PRINT TAB (40); "CORRECT (Y/XX)";
780 Y=0
800 INPUT X
820 IF X>15 GOTO 770
840 IF X>0 GOTO 400
880 Y=A(K+1)
900 CLS
920 FOR I = 1 TO Y
940 PRINT AT 11+I*6; I;
960 NEXT I
980 PRINT AT 64; "NET INCOME";
1000 Y=A(K+1)
1040 H=A(K+3)
1060 FOR I = 1 TO Y
1080 H=H*(1+A(K+2)/100)
1320 A(I)=INT(H+.5)
1340 PRINT TAB (10+I*6); A(I);
1360 NEXT I
1400 A=K
1450 B=K
1600 E=1+A(A+6)/1200
1620 L=1
1640 FOR I = 1 TO A(A+5)*12
1660 L=L*E
1680 NEXT I
1690 L=1/L
1700 A(B+16)=A(A+4)*A(A+6)/1200/(1-L)
1800 IF A=K A=A+3: B=B+1: GOTO 1600
1850 A=K
1870 B=K
1900 PRINT: PRINT "INTEREST"
1940 PRINT "  FIRST MTG";
2000 H=A(A+4)
2020 FOR I = 1 TO Y
2040 E=0
2060 J=(B-K+1)*Y+I
2080 IF I>A(A+5) E=0: GOTO 2250
2100 FOR L = 1 TO 12
2120 C=H*ABS(A(A+6))/1200
2140 E=E+C
2160 H=H-A(B+16)+C
2180 NEXT L
2250 A(J)=INT(E+.5)
2300 PRINT TAB (10+I*6); A(J);
2400 NEXT I
2420 A(B+18)=H
2440 IF H<0 A(B+18)=0
2500 IF A=K A=A+3: B=B+1:PRINT:PRINT"
     SECOND MTG";: GOTO 2000
2600 PRINT: PRINT "DEPRECIATION";
2630 E=A(K+11)
2650 L=A(K+11)/A(K+10)
2670 D=0
2700 FOR I = 1 TO Y
2720 H=E*A(K+12)/100/A(K+10)
2740 E=E-H
2760 H=INT(H+.5)
2800 A(Y+I)=A(Y+I)+A(Y*2+I)+H
2820 D=D+H-L
2840 PRINT TAB (10+I*6);H;
2860 NEXT I
3000 PRINT: PRINT "TAXABLE";
3040 FOR I = 1 TO Y
3060 A(Y+I)=A(I)-A(Y+I)
3100 PRINT TAB (10+I*6); A(Y+I);
3150 NEXT I
3200 PRINT: PRINT "TAX BENEFIT";
3260 FOR I = 1 TO Y
3270 H=-A(Y+I)*A(K+13)/100
3300 A(Y+I)=INT(H)
3340 PRINT TAB (10+I*6); A(Y+I);
3360 NEXT I
3400 PRINT: PRINT: PRINT "NET INCOME";
3500 FOR I = 1 TO Y
3520 PRINT TAB (10+I*6); A(I);
3540 NEXT I
3550 PRINT: PRINT "TAX BENEFITS";
3560 FOR I = 1 TO Y
3580 PRINT TAB (10+I*6); A(Y+I);
3590 NEXT I
3600 PRINT
3620 PRINT "MTG PAYMENTS"
3640 PRINT "  FIRST";
3700 A=K
3750 B=K
3800 H=INT(A(B+16)*12+.5)
3820 FOR I = 1 TO Y
3840 IF I>A(A+5) H=0
3850 IF I=A(A+5) H=H+INT(A(B+18)): A(B+18)=0
3900 PRINT TAB (10+I*6); -H;
3950 A(Y+I)=A(Y+I)-H
3970 NEXT I
4000 IF A=K A=A+3: B=B+1: PRINT: PRINT "  SECOND";: GOTO 3800
4100 PRINT: PRINT "CASH FLOW";
4140 FOR I = 1 TO Y
4160 A(I)=A(I)+A(Y+I)
4200 PRINT TAB (10+I*6); A(I);
4300 NEXT I
4400 PRINT
4450 PRINT TAB (20); "'ENTER' WHEN READY ";
4470 INPUT A$
4500 CLS
4520 PRINT AT 20; "FINAL SALES INFORMATION"
4550 PRINT
4600 E=1+A(K+2)/100
4620 L=1
4640 FOR I = 1 TO Y
4660 L=L*E
4680 NEXT I
4700 H=INT(A(K+14)*L)
4750 PRINT TAB (10); "SALES PRICE"; TAB (40); H
4800 E=INT(A(K+18)+A(K+19))
6000 PRINT TAB (10); "OUTSTANDING PRINCIPAL"; TAB (40); E
6100 B=INT(H*.05+.5)
6150 PRINT TAB (10); "COMMISSION (5%)"; TAB (40); B
6200 E=H-B-E
6250 PRINT TAB (10); "EQUITY"; TAB (40); E
6300 C=INT(A(K+14)-A(K+4)-A(K+7)+.5)
6600 PRINT TAB (10); "DOWN PAYMENT"; TAB (40); C
6700 B=H-B-A(K+14)+A(K+11)/A(K+10)*A(K+1)
6720 T=A(K+13)*.5/100
6750 IF T>.25 T=.25
6800 L=INT(T*B+.5)
6850 IFL>12 L=INT(12.5+(B-50)*A(K+13)/100+.5)
6900 PRINT TAB (10); "CAPITAL GAINS TAX"; TAB (40); L
7000 B=INT(D*A(K+13)/100+.5)
7050 PRINT TAB (10); "DEPRECIATION RECAP."; TAB (40); B
8000 A(Y)=A(Y)+E-L-B
8100 E=A(K+15)/100
8200 H=0
8220 L=1
8240 FOR I = 1 TO Y
8250 L=L*(1+E)
8260 H=H+A(I)/L
8300 NEXT I
8400 H=INT(H-C+.5)
8420 PRINT: PRINT: PRINT
8500 PRINT TAB (10); "NET PRESENT VALUE ";H
8550 PRINT
8600 PRINT
8620 PRINT TAB (20); "'ENTER' WHEN DONE ";
8640 INPUT A$
8660  GOTO 600
```

Photo courtesy Centuri Engineering Co.

Model Rocketry for Computer Hobbyists

BY G. HARRY STINE

Model rocketry is a natural hobby for personal computerists. No other hobby is so amenable to computation, so illustrative of Newtonian physics and so rife with computational problems both simple and complex. Simple computers witnessed the birth of the hobby twenty years ago, and model rocketeers have eagerly grasped and put to use each new step in computer technology.

Many people labor under the delusion that model rocketry is "kid stuff", an activity engaged in only by teenage boys who make some smoke in the sky for a few months before they move on to something else.

This delusion is like saying that once you've seen one computer, you've seen them all because they are nothing more than electronic adding machines. For model rocketeers there is always something new to learn, always a new experiment to make and always a new piece of research or development to do. After a physics degree and twenty years of adult life involved with model rocketry, I can truthfully say I've learned more about the way the universe works from model rocketry than I ever did from a physics professor.

But, back to square one. Model rocketry is space technology in miniature. In city parks every weekend you can do what they occasionally do at Cape Canaveral, but you can do it for only a couple of bucks. And, if you're a computer buff, you'll find all sorts of applications because, as miniature space technology, model rocketry has as much need for computers as the real thing. And you know what the computer has meant to space technology!

Model rocketry is also model aeronautics brought into the Space Age. It features lightweight, non-metallic, reuseable models made by the hobbyist from paper, cardboard, balsawood or plastic. Solid-propellant model rocket motors propel these models through the air. But no mixing or handling of toxic or explosive chemicals occurs because model rocket motors are one-shot, factory-loaded, replaceable units with paper casings slipped into the rear end of model rocket airframes. Model rocket motors contain everything necessary to thrust the model into the sky and to deploy the parachute or other recovery device that brings the model safely back to the ground, ready for another flight once the parachute is repacked and a new motor installed.

Following the commonsense Safety Code makes model rocketry one of the safest hobbies extant. In over twenty years, one hundred million model rockets have been flown in the United States with no injuries more serious than a few burned fingers and a few minor grass fires, which happened because someone ignored the hobby's simple safety rules.

Nor is model rocketry an expensive hobby. Less than \$20 gets you a starter set that includes everything you'll need except an ignition battery. Model rocket kits cost anywhere from \$1.50 to more than \$20, depending upon their size and complexity. The one-shot solid-propellant model rocket motors cost about 50 cents for each flight.

A typical model rocket is easy to assemble; seven-year-old children have done it successfully. Construction time runs from under thirty minutes to several hours, depending upon the model's complexity. You don't need special and expensive tools, only simple model airplane tools such as a sharp knife, sandpaper and glue.

Most small model rockets propelled by low-power motors can be flown safely from school football fields. The shortest dimension of the flying field should be no less than one-fourth the vertical altitude that you expect the model to achieve. Always use a launch pad. The launch pad rod provides initial guidance for the aerodynamically-stabilized model, permitting it to gain flying speed before leaving the launch rod. Simple electrical means, usually a system of fail-safe switches hooked to a car battery, ignite the model rocket motor.

Knowing weights, motor power and

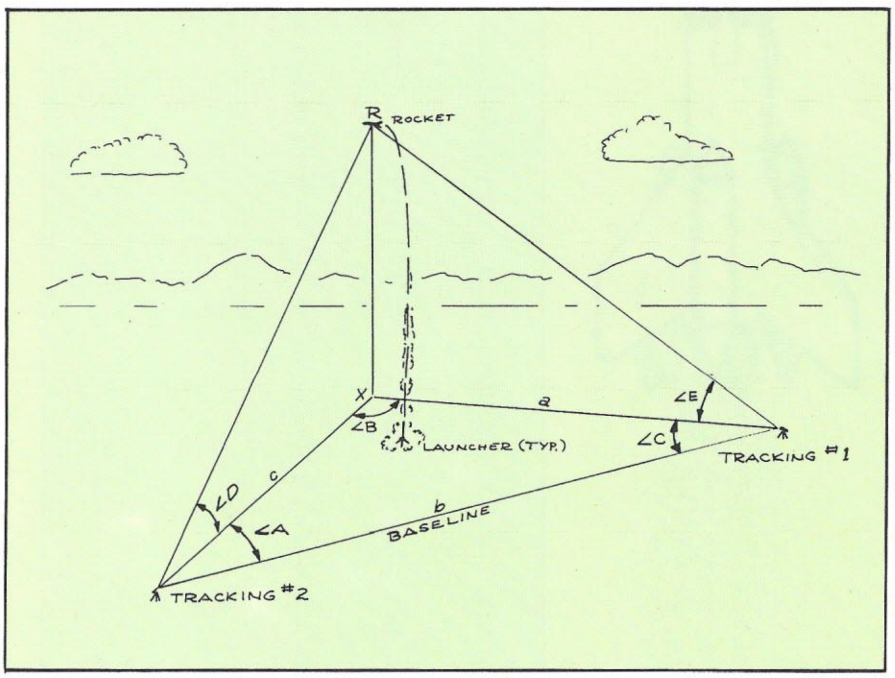
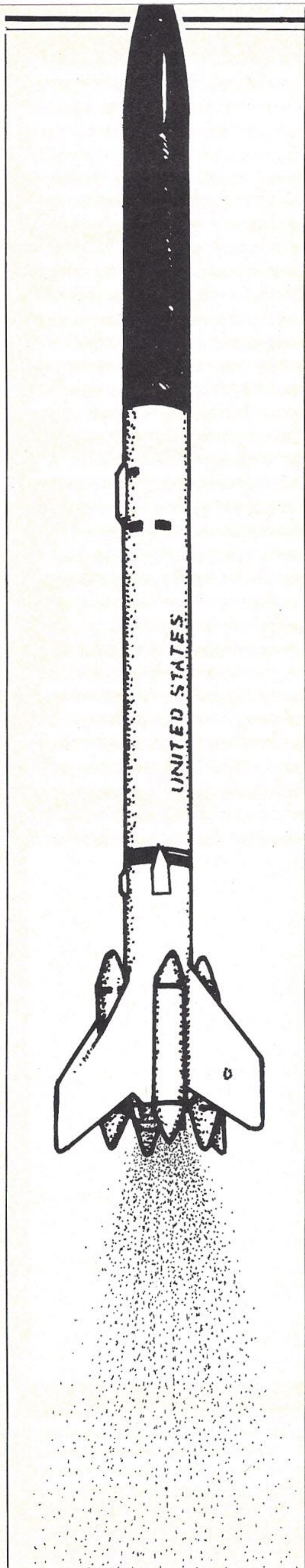


FIGURE 1 – You can determine your model rocket's achieved altitude using two trackers on a measured baseline. Two persons track the rocket to the peak of its flight and record the azimuth and elevation angle at that moment. Finding the altitude from the measured baseline and from the angles becomes a problem in trigonometry. (Drawing courtesy Centuri Engineering Company)



Program Notes

Press BST in RUN mode, switch to PRGM mode, then key in the program.

Registers:

R1: Elevation angle, Tracker No. 1
R2: Azimuth angle, Tracker No. 1
R3: Elevation angle, Tracker No. 2
R4: Azimuth angle, Tracker No. 2
R5: Baseline length (meters or feet)
R6: 180°
R7: Inter, Answer.
R8: No. 1 Altitude
R9: No. 2 Altitude

Note: All angles in decimal degrees. If baseline entered in meters, altitudes displayed in meters. If baseline entered in feet, altitudes displayed in feet. To check for "closed track," use standard deviation (gs).

Program Listing

DISPLAY LINE	CODE	KEY ENTRY	
00	---	----	
01	32	G	
02	44	Cl.R	
03	34	RCL	
04	05	5	
05	34	RCL	
06	06	6	
07	34	RCL	
08	02	2	
09	51	-	
10	34	RCL	
11	04	4	
12	51	-	
13	31	f	
14	12	SIN	
15	81	÷	
16	33	STO	
17	07	7	
18	34	RCL	
19	04	4	
20	31	f	
21	12	SIN	
22	34	RCL	
23	01	1	
24	31	f	
25	14	TAN	
26	71	X	
27	71	X	
28	33	STO	
29	08	8	
30	84	R/S	#1 Altitude
31	11	Σ+	
32	34	RCL	
33	02	2	
34	31	f	
35	12	SIN	
36	34	RCL	
37	03	3	
38	31	f	
39	14	TAN	
40	71	X	
41	34	RCL	
42	07	7	
43	71	X	
44	84	R/S	#2 Altitude
45	33	STO	
46	09	9	
47	11	Σ+	
48	31	f	
49	33	s	Average Altitude

aerodynamic factors, you can calculate the altitude that a given model rocket will fly to within one percent.

Predicting the flight path of a model rocket design lends itself to personal computers. A model rocket in flight is very Newtonian: a free body in space, interacting with Earth's gravity field and atmosphere, and no torque to worry about. Equations of motion of a model rocket in flight take all these factors into account.

There are two flight phases: powered and coasting. During powered flight the model rocket is under thrust from its motor. Known as a result of statistical static testing, motor thrust can be considered constant for simple calculations. That thrust changes in known and predictable fashion during flight brings in other variables that keep advanced theoretical people very happy. During coasting flight the model trades vertical velocity for altitude as a free-falling body in a gravity field.

Because you can determine such variables as thrust, total impulse, propellant burn rate, liftoff weight, burnout weight and aerodynamic drag, you can compute altitude performance. The equations are simple if aerodynamic drag is ignored . . . which you cannot do because drag is a major factor affecting the flight. Before the introduction of small, multi-function hand calculators and microcomputers, determining anticipated altitude with aerodynamic drag taken into account required several hours of laborious hand calculation by the iterative method.

Aerodynamic drag is a function of air density (there is less air drag on a hot day at high altitude), velocity (a varying and exponential function), the model's frontal area (big models have more drag than small ones), and a non-dimensional number called "drag coefficient", a function of the model's shape and surface finish. It took hours to do this messy calculation by hand twenty years ago. Ten years ago, it required a UNIVAC Thin-Film Memory 1107 digital computer, FORTRAN programming and 13,500 separate calculations. Today, it can be done on most programmables.

Altitude tracking, or, in other words, "How high did it go?" is easier to perform. Over the past twenty years, many methods have been tried for determining the maximum altitude achieved by a model rocket, including radar. The best and most accurate – and the method used for both national and international

competition where achieved altitude is scored – is simple triangulation using two trackers.

A typical altitude tracking device measures a horizontal azimuth angle and a vertical elevation angle, like a theodite, but not as accurately. A model rocket tracker need only be accurate to within a half-degree. Trackers are simple to make and can be mounted on ordinary camera tripods.

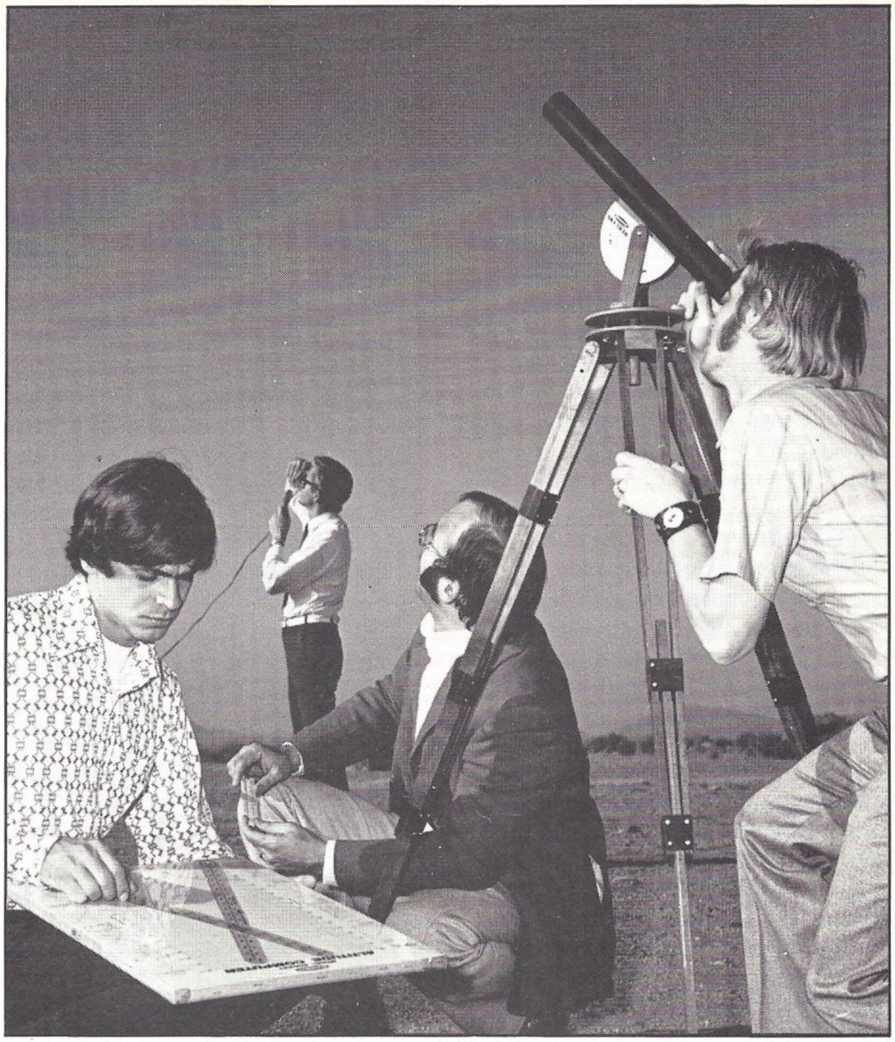
Using two trackers on a known, measured baseline as shown in Figure 1, you can accurately determine the achieved altitude. The baseline is normally 300 meters (984.24 feet) long, an international standard, but it can be longer for increased accuracy. Two trackers are set up at each end of the measured baseline. The azimuth and elevations dials of both trackers are set to zero when each tracker looks directly at the other.

The launcher can be anywhere, *except* on or near the baseline because of inaccuracies in the data management system for small azimuth angles. Normally, the system is set up so that each tracker has about a 30-degree azimuth reading when looking at the launcher.

Operating this tracking system takes two people with sharp eyes. It also helps if the model rocket is painted a bright flourescent orange (for clear skies) or flat black (for cloudy skies). Telephone or CB walkie-talkie communications are usually set up between trackers and launcher so that the trackers will know when the model is to be launched.

When the model rocket is launched, the tracking station people follow it visually and, when the model slows down near the peak of its flight, they zero in on it with their trackers. At the peak of the flight, they stop tracking. The recovery device then blossoms forth to lower the model to a gentle landing. Each tracking station reports or records the azimuth and elevation angle of the peak point in the model's flight. We now have a trigonometry problem: finding the vertical distance between the model and the ground from one measured distance (the baseline) and four angles. But with a small hand calculator right on the flying field you can determine the achieved altitude right away.

Basically, the azimuth angles determine an imaginary horizontal triangle that locates the sub-model point on the ground. The elevation angles determine two vertical triangles with a common side, RX. The achieved altitude RX can be computed twice for redundancy using each of the two vertical triangles. RX as computed from both elevation angles should agree within an accepted margin of tolerance. Usually, the altitude is taken as the average of the two computed altitudes, and each of the two computed altitudes must be within 10% of the mean altitude for the track to be considered valid.



Altitude determination by triangulation is used in national and international competition and for establishing records. Early tracking (shown here) used simple analog computers. Now, data reduction of angles is done with hand calculators. (Photo courtesy Centuri Engineering Company)

Referring to Figure 1 and without going into the derivation of the equation (the simple proof is left to the reader as an exercise, or you can find it in my *Handbook of Model Rocketry*), the achieved altitude RX from Tracking No. 1 can be expressed:

$$RX_1 = \sin\angle C \tan\angle D \frac{b}{\sin[180 - (A+C)]}$$

Similarly, for Tracking No. 2:

$$RX_2 = \sin\angle A \tan\angle E \frac{b}{\sin[180 - (A+C)]}$$

The two computed altitudes are then averaged:

$$\frac{RX_1 + RX_2}{2} = RX$$

This sort of problem can neatly be set up with any hand calculator having trig functions. If you don't have one, use your computer to work out a table with sin and tan functions of each angle, plus a column of precalculated $\frac{b}{\sin[180 - (A+C)]}$. If you have a programmable calculator, you can set up your own program ahead of time and then merely enter the four angles in proper sequence.

A program for the Hewlett Packard HP-55 is shown in the Program Listing. Once you've entered the program into the calculator, most of the work is finished. The next step is to enter into the registers the angular data furnished by the two tracking stations, plus the baseline length data. With all informa-

tion keyed in, switch to run mode. When R/S is pushed the first time, it will display the first computed altitude. Push R/S again, and the second altitude will be displayed. The third activation of R/S displays the average altitude. At this point, the HP-55 has run out of program steps, but the standard deviation to check for track closure can be obtained by pressing gs.

Entering this program the first time seems like a lot of work. But it pays off when you are standing in the middle of a field with tracking data coming in. You can obtain a number for achieved altitude before your model rocket touches the ground!

This example is but one of the uses of computers in model rocketry. Small personal computers can also determine aerodynamic characteristics such as stability, center of pressure and center of gravity. Trajectory analysis and design analysis are two applications. When you get into something as complex and advanced as dynamic flight stability (which even professional sounding rocket engineers don't fully comprehend yet!), computers are an absolute requirement because of the iterative nature of the solutions.

Model rockets are more than just up-and-down toys. Students at such places as MIT and the USAF Academy have applied large, multi-purpose computers to problems in model rocketry because these problems are analogous to those in astronautics and can be field-checked with inexpensive models. Model rocketry was born with the Space Age and the Computer Age twenty years ago, and it is growing up with computers. Model rocketry is a technology in miniature, and it is therefore not surprising to find it intimately linked with personal computers. □

* * *

FOR MORE INFORMATION:

The Handbook of Model Rocketry, G. Harry Stine, Follett Publishing Company, 1010 W. Washington Blvd., Chicago, IL 60607, $6.95 paperbound.

The National Association of Rocketry, P.O. Box 725, New Providence, NJ 07974.

Centuri Engineering Company, P.O. Box 1988, Phoenix, AZ 85001.

Estes Industries, Inc., P.O. Box 227, Penrose, CO 81240.

Flight Systems, Inc., 9300 E. 68th St., Raytown, MO 64133.

Model Rocket Altitude Program

BY GEOFFREY A. LANDIS

Here is a simple but accurate model rocket altitude program containing the basics of almost any altitude program. You'll surely want to add your own refinements, such as provisions for more or different rocket engines (I arbitrarily picked a C6 engine for this program), variable air density, multi-staging, mass loss during burn, or even two-dimensional trajectories (e.g., range as well as altitude). You could also add a more accurate simulation algorithm.

The program works straightforwardly, so beginning as well as advanced rocketeers can follow. The program's first section initializes variables to start the rocket off on the ground and sets constants such as engine burn parameters. The heart of the program simply runs a continuous loop, each time around incrementing time by t1, calculating engine thrust, and, from thrust, acceleration, velocity and distance traveled. The only unfamiliar equation here will be the law of drag:

$$\mathrm{DRAG} = \tfrac{1}{2}\rho C_d A\, V^2$$

Lacking Greek symbols and subscripts, I called ρ (rho, aerodynamicists' symbol for air density) the letter "R"; and C_d (Drag Coefficient, a measure of how much resistance the model meets in the airstream) I called C1. To save the computer loop time, the constant part of this expression is calculated beforehand. A is frontal area, equal to πr^2 or $\frac{1}{4}\pi d^2$; where d is your model's diameter.

Calculation of engine thrust I put into a subroutine for two good reasons. First, you may want to put in capability to choose one of several available rocket engines – I doubt that you always fly with C6s. This capability can be added with a computed GOSUB statement. Second, you may wish to make a much more complicated simulation of the engine. For simplicity I just used

Sample Run

```
Do you want to change any values? yes
Input rocket mass in kg: .20
Input maximum body diameter in meters: .04
Input Drag Coefficient: .85
time            altitude          velocity        acceleration
 0.05            0.08795           1.759           35.18
 0.1             0.263825109849    3.51750219698   35.1700439396
 0.15            0.527550687267    5.27451154837   35.1401870279
 0.2             0.879002464548    7.02903554562   35.090479945
 0.25            1.28050678741     8.03008645718   20.0210182313
 0.3             1.73194238632     9.02871197828   19.9725104218
 0.35            2.23317222142     10.0245967019   19.917694472
 0.4             2.78404365024     11.0174285765   19.856637492
 0.45            3.38438861483     12.0068992918   19.7894143055
 0.5             4.03402384755     12.9927046545   19.7161072541
 0.55            4.73275109523     13.9745449536   19.6368059827
 0.6             5.48035736093     14.952125314    19.5516072071
 0.65            6.27661516278     15.9251560371   19.4606144638
 0.7             7.12128280925     16.8933529294   19.3639378448
 0.75            8.01410469001     17.8564376152   19.2616937168
 0.8             8.95481158184     18.8141378366   19.1540044268
 0.85            9.94312096866     19.7661877364   19.040997996
 0.9             10.978737375      20.7123281265   18.9228078021
 0.95            12.0613527119     21.6523067391   18.7995722518
 1               13.190646635      22.5858784613   18.6714344453
 1.05            14.3662869127     23.512805553    18.5385418337
 1.1             15.587929805      24.4328578465   18.4010458702
 1.15            16.8552204514     25.3458129294   18.2591016577
 1.2             18.1677932669     26.251456309    18.112867593
 1.25            19.5252723449     27.1495815595   17.9625050094
 1.3             20.9272718674     28.0399904505   17.808177819
 1.35            22.3733965203     28.9224930583   17.650052156
 1.4             23.8632419133     29.7969078594   17.488296022
 1.45            25.3963950036     30.6630618061   17.3230789347
```

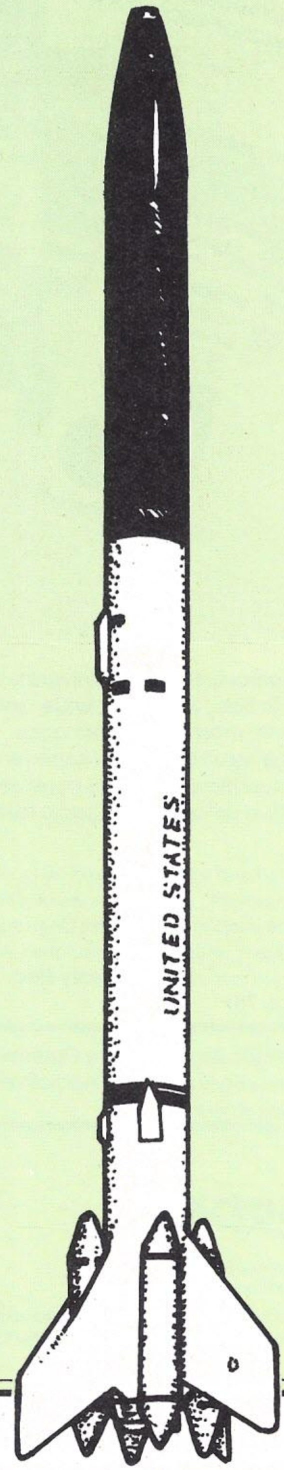
two main parameters right out of the manufacturer's specs – the thrust spike (F1) and the plateau thrust (F2). Burn time of 1.7 seconds also comes from the specs. A simpler approximation, virtually as good, assumes the engine has a constant thrust, equal to average engine thrust, for duration of the burn and zero thereafter. For a more complicated simulation, choose several data points out of the manufacturer's published time-thrust curves and linearly interpolate between them. □

Program Listing

```
10 REM****************************************************************
20 REM        MODEL ROCKET ALTITUDE SIMULATION PROGRAM
30 REM               program by Geoffrey A. Landis
40 REM                          1978
50 REM****************************************************************
60 T=0
70 V=0
80 X=0
90 G=9.82
100 C1=0.75
110 D=0.02
120 R=1.205
130 M=0.1
140 T1=0.05
150 T2=0.2
160 T3=1.6
170 F1=9
180 F2=6
190 PRINT "Do you want to change any values? ";
200 INPUT A$
210 IF A$="Y" OR A$="YES" THEN 1090
220 C=R*PI*C1*D*D/8/M
230 PRINT "time              altitude         velocity        acceleration"
240 REM altitude loop starts here
250 T=T+T1
260 GOSUB 1000
270 A=F/M-G-C*V*V
280 V=V+A*T1
290 X=X+V*T1
300 IF V<0 THEN 330
310 PRINT T,X,V,A
320 GO TO 240
330 PRINT "  MAXIMUM ALTITUDE ATTAINED IS ";X;" METERS "
340 END
```

```
1000 REM ENGINE THRUST SUBROUTINE ************************************
1010 REM   this subroutine is a simple approximation to the thrust of
1020 REM   a C6 type model rocket engine
1030 F=0
1040 IF T>T3 THEN 1080
1050 F=F2
1060 IF T>T2 THEN 1080
1070 F=F1
1080 RETURN
1090 REM PROCEDURE FOR MANUAL CHANGES OF ROCKET PARAMETERS
1100 PRINT "Input rocket mass in kg: ";
1110 INPUT M
1120 REM  t2 is the time duration of the peak thrust in seconds
1130 REM  t3 is the total burn time of the engine in seconds
1140 PRINT "Input maximum body diameter in meters: ";
1150 INPUT D
1160 PRINT "Input Drag Coefficient: ";
1170 INPUT C1
1180 GO TO 220
```

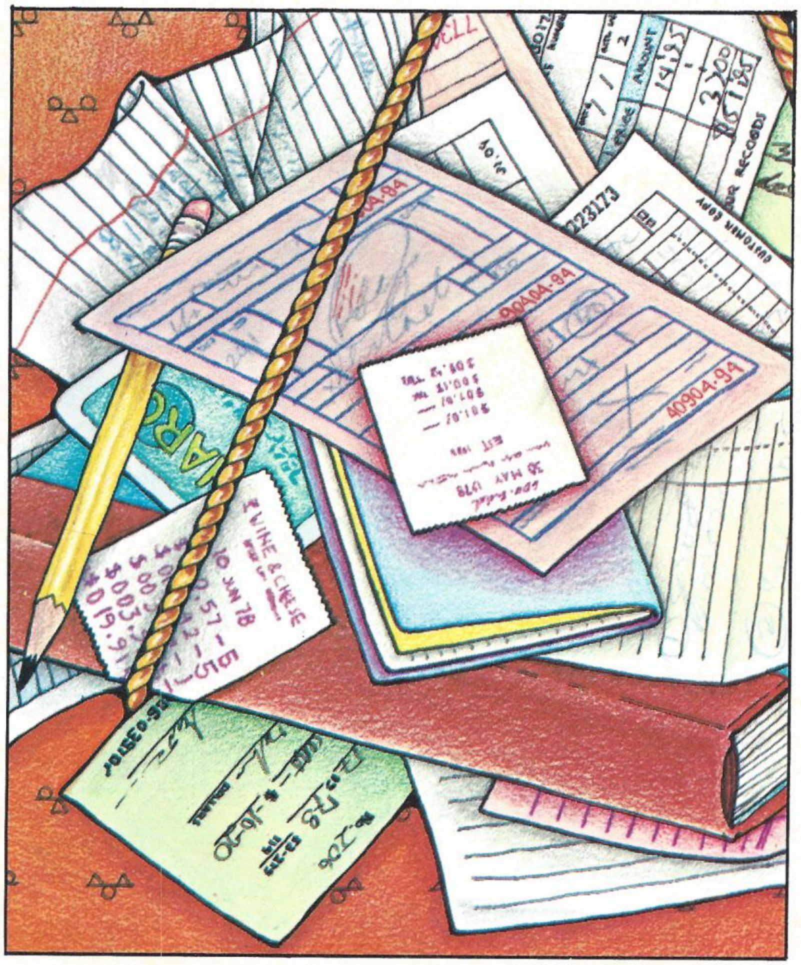
```
2000 REM        VARIABLE NAME LIST
2010 REM          all units are MKS metric, of course.
2020 REM   a  is the rocket acceleration in meters per second per second
2030 REM   c  is a intermediate result used in calculating drag
2040 REM   c1 drag coefficient of rocket
2050 REM   d  diameter of the rocket in meters
2060 REM   f  is the engine thrust at time t
2070 REM   f1 is the thrust of the rocket engine at peak, in newtons
2080 REM   f2 is the average thrust of the rocket engine, in newtons
2090 REM   g  acceleration of gravity in meters per second per second
2100 REM   m  rocket mass in kilograms
2110 REM   R  density of air in kilograms per cubic meter
2120 REM   t  time from ignition in seconds
2130 REM   t1 time increment between passes through loop in seconds
2140 REM   v  rocket velocity in meters per second
2150 REM   x  rocket altitude in meters
```



The Computer Checks into the Balancing Act

BY O.E. DIAL

PART III



CRAS is a fully integrated Check Register based Accounting System. The files established when a transaction entry is made – think of them as checkbook entries with more detail – support a wide variety of printed statements. Parts I and II (August and October *PC*) listed the first 9 programs together with user instructions. This final article in the series provides the remaining three. A later article will integrate the Recursive Budgeting model (May/June 1977) into the CRAS system.

CRAS originally consisted of a single program which accepted transaction entries and printed account distributions and statements of account. The remaining programs now in CRAS were added one at a time, each considerably increasing the core memory requirements for a run. This situation explains the original file structure, numbering from 1 to 11.

Buffer space reservations during initialization for random files is just under 400 bytes. When CRAS was completed, some 4400 bytes of core memory were taken off the top during initialization. My Altair, with 60K bytes of memory, posed no problem. But memory requirements might easily pose a problem to other systems with less core. Yet, in segmenting the system into 12 programs, I retained the original file numbers.

If you need to be more economical with core requirements, you may go through each program, numbering the various files sequentially within the same program. You'll find a maximum of only 8 file buffers are required, thus saving 1200 bytes of core at the outset.

A remaining preliminary matter – some alterations of the Transaction Entry Program (CRAS-TE) crept into the final listing before I sent it to *Personal Computing.* The corrections to the program are listed in Table I.

CRAS-SF (Program 10)

This Suspense File program creates a suspense file which is limited to ten reminders per month. The file may be edited, and suspense items may be added or deleted at any time. The program automatically arranges supense items within the same month in chronological order. By selecting option 4 from the menu, the program ends and CRAS-RS (Routine Select) is run.

Most of the coding techniques used in the program were described in earlier articles in this series. But a few merit discussion. For example, see lines 700 to 720. The file fielded in line 700 reserves two bytes for the day of the month (an integer, although you could easily use a string), and reserves the remainder of the field for the message string.

Notice that in line 720 the record number is established by 10*J+K, where "J" is the month and "K" the message number within the month. The "J" loops run from zero to eleven. Thus, where the "J" is to indicate the month, 1 is added to it to reconcile with the calendar. See statement 860.

Statements 880 through 900 exhibit variable-length caption underscoring. In statement 940, the date field of each record is used to indicate the end

Continued on Page 72

Acronym Glossary

BY DAVE POWELL

Confused by all the LEDs, LSIs *and* CRTs *you see in computer literature? Can't tell a* RAM *from a* ROM *– or either from a* PROM*? Ever wondered why* BASIC *is called* BASIC *and* APL, APL*? And what* are *bits, bytes and nibbles, anyway?*

Use this handy acronym glossary to take the mystery out of those strange computer terms.

Dave Powell is Associate Editor of PC*'s sister publication,* Minicomputer News, *where this article first appeared.*

A/D, D/A (Analog-to-Digital, Digital-to-Analog converter) – A device that changes physical motion or electrical voltage into digital signals consisting of ones and zeros, and vice-versa.

ALGOL (Algorithmic Language; Algorithmic Oriented Language) – A computer programming language used to code problem-solving formulas and logic according to internationally recognized standards. The language also allows standardized communication of arithmetic procedures among individual programmers.

ALU (Arithmetic and Logical Unit) – The portion of computer hardware that performs arithmetic and logical operations. Part of a computer's CPU, this unit performs when instructed by the CPU's Control Unit.

APL (A Programming Language) – A language with a syntax and character set designed for mathematical applications, especially those requiring numeric or literal array manipulation.

ASCII (American Standard Code for Information Interchange; abbreviation pronounced "ass-key") – Synonymous with USASCII, this code equates input/output device-control commands, alphanumeric and special characters with standardized 7-bit sequences. For example, in ASCII code, "1" is "0110001", "A" is "1000001" and "$" is "0100100". This standardized communication code allows different system and peripheral manufacturers to build communications-compatible hardware.

BASIC (Beginners All-purpose Symbolic Instruction Code) – A language developed at Dartmouth College that is easy to learn due to its small repertory of commands and simple statement formats. For this reason, BASIC is widely used in programming instruction, in hobby computing, in business and in industry.

BCD (Binary Coded Decimal) – A technique for handling decimal values, especially in COBOL and assembly programs. With BCD coding each decimal digit is equivalent to four binary bits.

Bit (Binary Digit) – Within computer storage all program statements and data are represented as strings of bits, ones and zeroes. User programs submitted for execution must be converted to bit strings (called machine language) before the computer can act on program statements. Bits exist within computer hardware in the form of circuits that are carrying current (1) or are not (0) or in devices that are magnetized or not. Small groups of adjacent bits (usually from four to eight), when used by the computer as a unit, are called bytes. Half a byte is called a nibble, honest.

CAD/CAM (Computer-Aided Design/Computer-Aided Manufacture) – Computer-aided design programs allow part designers to both structure and test parts and assemblies before manufacture. Computer-aided manufacture computers support employees and equipment during the manufacturing process.

CAI (Computer-Aided Instruction) – The step-by-step, interactive process where the computer leads the student through concepts to be learned.

CCD (Charge-Coupled Device) – A recently introduced solid-state memory device within which stored data circulates, rather than remaining in fixed locations (as data does in RAMs, ROMs, PROMs, etc.). Data to be read is "siphoned off" as it passes a read location, called a "drain". Data feeds into the circulating stream through a "source". The name Charge-Coupled comes from the driving force behind data circulation: that opposite charges attract and like charges repel. CCDs tend to be slower than RAM devices, but provide higher data-storage densities and lower cost per bit stored.

CMOS (Complimentary MOS) – Hybrid devices combining both PMOS and NMOS technologies that operate as fast as NMOS chips, but consume less power.

COBOL (Common Business Oriented Language) – A programming language developed to express problems to the computer using common English nouns, verbs and connectives.

CPU (Central Processing Unit) – The part of the computer containing circuits to interpret and execute program statements, including logic, arithmetic and input/output statements. The CPU also contains small (relative to disk) but quickly accessible "main" or "central" storage hardware.

CROM (Control Read-Only Memory) – A single read-only chip (see ROM) that contains all programming to control a manufacturer's microprocessor. Because these programming instructions are common to all similar micros, CROMs are incorporated into peripheral hardware to test and develop programs before attempting to run them on the microprocessor.

CRT (Cathode-Ray Tube) – A vacuum tube, much like a television screen, within which a beam of electrons is fo-

cused to a small spot on a luminescent screen and deflected magnetically to trace output graphics. Some CRTs, called storage tubes, can hold images on the screen continuously without electron-beam retracing.

CT (Computed Tomography) – Formerly called CAT (Computerized Axial Tomography). A medical application in which a computer records X-rays passing through the body in changing directions and generates an image of the body's structures. Recent advances in nuclear medicine allow CT scanners to image both body structures and chemical composition.

DIP (Dual In-Line Package) – Integrated circuit chips are so small that attaching wires to them is difficult. So chips are mounted in plastic cases with side-mounted electrical leads connecting the chip to outside circuitry. Because DIPs resemble flat caterpillars, they are often called "bugs".

DP (Data Processing) – A collage of two terms in today's popular usage: "computing" and "data processing". Computing means setting up procedures for obtaining numerical output. Data processing usually involves such manipulations as storage, retrieval, sorting, merging, updating, assembling, searching, reading, writing and erasing.

DTL (Diode-Transistor Logic) – Microelectronic logic based on connections between semiconductor diodes and transistors. DTLs improved performance of earlier TRL devices by substituting diodes for many resistors. (Also see TTL, RTL, TRL.)

EAROM (Electrically Alterable Read-Only Memory) – ROM memory that can be selectively altered without erasing all stored data, as is done with EPROM devices.

EBCDIC (Expanded Binary-Coded Decimal Interchange Code, abbreviation pronounced "ebb-sih-dick") – An eight-bit code used to represent 256 letters, numbers and characters. EBCDIC is commonly used in IBM computers.

EDP (Electronic Data Processing) – Data processing (see DP) assisted by computer, sometimes without human intervention (unattended operation).

EFT (Electronic Funds Transfer) – Often referred to as the beginning of "The Cashless Society", EFT networks transfer money among businesses and banks via computers, rather than paper media such as checks.

EPROM (Electrically Programmable Read-Only Memory, also called Erasable Programmable Read-Only Memory) – Like most modern solid-state memory systems, EPROMs are manufactured by "etching" microscopic circuit paths into silicon wafers and cutting the wafers into "chips" smaller than one's fingertip. EPROMs are, first, ROMs, meaning that during normal computer operation, information stored on the chip, such as pocket calculator control programs, may only be read and executed, but not altered. However, EPROMs are also erasable; the chip may be wiped clean of data by exposure to ultraviolet light. Then the chip is reprogrammed electronically.

FIFO (First-In-First-Out) – A method of building and accessing data files where the first data entered is also the first retrieved. In other words, the next item removed from the file is the item that has been in the file longest. (Compare with LIFO.)

FORTRAN (Formula Translator) – A language developed by the International Business Machines Corp. FORTRAN allows programmers to code instructions very similar to ordinary mathematical equations. FORTRAN is a compiler language; compiler (translator) software converts each language statement directly into machine language (strings of ones and zeroes).

GIGO (Garbage-In-Garbage-Out) – The concept behind the belief that computers are no better than the people who program them. Technically, if meaningless, incoherent, wrong or nonsensical data is input to a computer (whether deliberately or accidentally), the same will come out.

IC (Integrated Circuit) – An electronic circuit made from a single piece of semiconducting material (usually silicon) by controlling the geometry and electrical conductivity of circuit paths. ICs are usually designed to perform a complete function, and cannot be disassembled without destroying them.

I/O (Input/Output) – A general term referring to equipment used to communicate with a computer, to the data communicated, to the media carrying the data or to the process of communicating with a computer.

K (as in 32K bits, 32K bytes or 32K words) – K, short for kilo, meaning 1000, is usually used in the computer industry to mean 1024. Therefore 32K usually means 32,768. K often measures storage capacity, either in bits (zeroes and ones), bytes (usually groups of eight bits) or words (usually two or four bytes). (Also see M.)

LED (Light-Emitting Diode) – A crystal that glows when electric current passes through it. A matrix of LEDs can be used to display numbers and letters by passing current through appropriate members of the matrix. Members are chosen by outside logic circuits.

LIFO (Last-in-First-Out) – A method of building and accessing data files where the last data entered is the first retrieved. In other words, the next item removed from the file is the item most recently entered. (Compare with FIFO.)

LSI (Large-Scale Integration) – The

Acronym Glossary

process of placing over 100 integrated circuits (ICs) on one silicon chip. (Also see MSI.)

M (as in 4M bits, 4M bytes or 4M words) – M, short for mega, meaning 1 million, is used in the computer industry to mean 1,048,576, the second power of 1024 (see K). As with K, M usually is used for storage measurements, although to date M most frequently applies to peripheral memory devices, such as disks. Recently introduced minicomputer systems do offer CPU memory capacity ranging into megabytes.

Macro (Macroinstruction) – A single programming language instruction that (1) generates a group of machine-language instructions (in the binary code understandable to the machine), (2) is replaced by a predefined set of machine-language instructions or (3) calls for prewritten special routines to perform desired functions during program execution.

MICR (Magnetic Ink Character Recognition) – A system developed by the American Banking Assoc. to allow computers to read and sort documents such as checks. Standardized characters are printed with a magnetic ink. Special scanners read the ink, which is also visible to the eye (along the bottom edge of a check, for example), the input allowing computers to sort, sum and route financial transactions. Often magnetic ink character recognition is used where accidental marks would confuse an Optical Character Recognition system (see OCR).

Modem (Modulator-demodulator) – A device that converts computer data to a form compatible with data transmission facilities, and vice-versa. Often, modem hardware is incorporated into compact data-entry-and-transmission terminals.

msec, ms (millisecond) – An abbreviation for one-thousandth of a second.

MOS (Metal-Oxide Semiconductor) – Part of the acronym MOSFET, meaning Metal-Oxide-Semiconductor Field-Effect Transistor. MOS devices are solid-state electronic circuits deposited in successive layers on a silicon base. Microcomputer memory and logic chips are based on MOS technology, while larger systems usually incorporate MOS chips in central memory alone. (Also see CMOS, MOS, NMOS, PMOS.)

MSI (Medium-Scale Integration) – The process of forming between 10 and 100 circuit components, such as transistors, on one silicon chip. (Also see LSI.)

MTBF (Mean Time Between Failures) – In slightly simplified terms, the average equipment operating time between hardware failures. MBTF is generally calculated as the limit of the ratio of operating time to the number of failures as the number of failures approaches infinity.

MTTF (Mean Time To Failure) – The average time between initial equipment start-up and the first failure or malfunction, calculated by measuring this time on many identical pieces of equipment.

MTTR (Mean Time To Repair) – The average time experienced by equipment users between an observed failure and corrective maintenance. MTTR includes the time required to reach the repair center and the time the center takes to respond.

NMOS, N-MOS (N-channel MOS circuits; see MOS) – Metal-Oxide Semiconductor circuits first developed in 1973, where the electrical current is a flow of negative charges (electrons). According to some sources, NMOS technology produces circuits that run twice as fast as PMOS-based circuits (see PMOS), are smaller than PMOS circuits, need less power and generate less heat. NMOS chips are frequently used in applications requiring high-density memory and logic circuits.

nsec, ns (nanosecond) – One-billionth of a second. Most computers' central processor units (CPUs) recycle so fast that the time required to begin executing another instruction is measured in nanoseconds. Nanosecond is sometimes abbreviated mμsec, since it is equivalent to one milli-microsecond.

OCR (Optical Character Recognition) – Machine identification of printed characters using light-sensitive materials or devices. The post office uses OCR scanners to read zip codes on mail. (Also see MICR).

OEM (Original Equipment Manufacturer) – A somewhat vague term, generally meaning a supplier that collect products from outside sources and assembles them into a final product or system for end users.

OS (Operating System) – A collection of read-only programs (normally not accessible to user-programmers) that assigns I/O hardware, central and peripheral storage and CPU processing time. Operating systems also control program execution order, especially in multiprocessing systems where more than one user shares the computer's time. The operating system is sometimes also called "system executive" or "system monitor".

PC (Printed Circuit) – A circuit whose elements, such as resistors, transistors, diodes and capacitors, are not joined by wires but by conducting channels printed onto a supporting card. This printing is done by covering the card with copper (the conductor), covering the copper with light-sensitive chemicals (similar to photographic emulsion), placing a mask over the card and exposing the mask and card to light. Where light passes through the mask and hits the card, a subsequent acid bath leaves the copper untouched. Copper not "fixed" by the light is eaten away. Copper left on the card represents conducting paths.

PIN (Personal Identification Number) – A security number computers sometimes require before a user can access the system or before a point-of-sale (POS) terminal customer can enter or receive money or data.

PL/1 (Programming Language/1) – A language developed by International Business Machines Corp. that incorporates FORTRAN's numeric manipulation capabilities and alphanumeric manipulation features, such as data structures, editing and string manipulation, from COBOL and ALGOL. These features tailor the language to semicommercial-semiscientific applications.

PMOS, P-MOS (P-channel MOS circuits; see MOS) – Metal-Oxide Semiconductor circuits (the earliest type), where the electrical current is a flow of positive charges (protons). PMOS technology was used in early Intel and Rockwell microprocessors, but is being replaced by NMOS or CMOS technologies, which produce faster, more power-efficient circuits. PMOS is still used in pocket calculator chips.

POS (Point of Sale terminal) – "Smart" terminals (containing microprocessors) designed to interact with a business' customers and/or staff. POS terminals often tie with a central computer and help perform calculations (such as total price), control keyboard access, update inventory, read product tags and accept credit-card input. Eventually POS networks may extend beyond home companies into state and national electronic funds transfer (EFT) systems – another move toward "The Cashless Society."

PROM (Programmable Read-Only Memory) – A MOS-technology-based read-only memory (ROM) that is not programmed for its future function during manufacture. The end user inserts machine-language programs via a PROM programming terminal. Some can be erased and reprogrammed.

psec, ps (picosecond) – One-millionth of one-millionth of a second. Equal to one-thousanth of a nanosecond (see nsec) or one micro-microsecond (see usec).

RAM (Random Access Memory) – Any storage device that will input/output data with the same rapidity, regardless of the data's location within the memory. Such devices as solid-state memory chips and, to some extent, moving disks offer random access to stored data. Others do not, such as cassette or reel tapes, which require the read/write mechanism to wait until all tape between the present location and the desired data passes by. Such devices are called sequential-access memory. In present usage, RAM almost always refers to LSI memory chips.

ROM (Read-Only Memory) – In present usage, a solid-state storage chip programmed for its future function at manufacture, and not reprogrammable by the user. Information on ROM may only be read. Applications include control programs for pocket calculators, microprocessors and minicomputers. (Also see EPROM, PROM for variations on a theme.)

RPG (Report Program Generator) – A language or complete program that generates computer code to manipulate and output data according to user-specified formats (in effect a program to generate programs). User input includes headings, styles and data locations.

RTL (Resistor-Transistor Logic) – Microelectronic logic circuits based on connections between semiconductor resistors and transistors, with the number of transistors maximized. This was the first logic incorporated into microelectronics devices. (Also see DTL, TRL, TTL.)

SNOBOL – A string-manipulation and pattern-recognition language developed by Bell Laboratories.

SOP (Standard Operating Procedure) – The status quo. The absence of Murphy's Law's effects.

SOS (Silicon on Sapphire MOS technology) – The process of fabricating integrated circuit (IC) chips on layers of silicon and sapphire.

Telex, TEX (Teletype Exchange) – A teletype communications service provided by Western Union, in the United States, and by the Canadian Pacific Railroad, that allows subscribers to direct-dial each other's teletype.

TTL, T^2L (Transistor-Transistor Logic) – The most common form of microcircuit logic based on connections between semiconductor transistor elements.

TRL (Transistor-Resistor Logic) – Microelectronic logic circuits composed of semiconductor transistors and resistors, with the number of resistors maximized to reduce fabrication cost.

TTY – Acronym for teletype.

TWX (Teletypewriter Exchange Service) – An American Telephone and Telegraph Service that connects subscribers teleprinter equipment with the public switched telephone network.

UPC (Universal Product Code) – Developed in 1973 by the supermarket industry for identifying products and manufacturers on product tags. A variety of manufacturers produce UPC printers to write the 10-digit optical-bar symbols and UPC scanners to read the codes during checkout.

usec, us, μs (microsecond) – One-millionth of a second.

WP (Word Processing) – The computer systems designed to handle words and text as input and output, rather than to perform scientific calculations or control realtime processes.

A Feast of Micro

No matter what application you have in mind, you still need to examine the same microprocessor features to choose the unit best suited to your needs. The authors evaluated a wide range of microcomputers to pick the best ones to use for teaching science students. But you can use their data and comments to help you select a microcomputer for your home or business.

BY MORRIS FIREBAUGH, LUTHER JOHNSON AND WILLIAM STONE

Students entering science and engineering careers will certainly use instruments incorporating microprocessors and microcomputers. Many will work in experimental design applications for which microcomputers provide a flexible prototype system. Physics laboratories often provide the first formal exposure to electronic instrumentation for such students. Thus, science teachers should provide students with hands-on laboratory experience in microcomputers.

We wrote this article, not to pass judgement on individual products, but to present an overview of the field to help science teachers select microcomputers for classroom and laboratory use. While we don't provide an exhaustive microcomputer catalog, the units discussed here together comprise a substantial portion of the microcomputer market.

Manufacturers advertising in major journals were invited to send units for evaluation, in kit form if possible. We assembled, inspected and tested the computers for normal operation and ran a benchmark program on each unit.

Criteria we believe are important in selecting a microcomputer for instructional purposes include microprocessor characteristics, communication ability (programming and debugging, I/O ports for interfacing and peripherals available), quality of parts, software system (flexibility and availability of applications software), quality and level of documentation and cost effectiveness for expected uses.

The units we examined could generally be classified into one of four main categories: (A) Mainframe, (B) Tutorial System, (C) "Black Box" complete personal computer and (D) Single Board Computer.

This article is reprinted with permission from the October 1978 issue of The Physics Teacher.

Opinions expressed in this article are those of the authors and do not necessarily reflect the views of Personal Computing *magazine.*

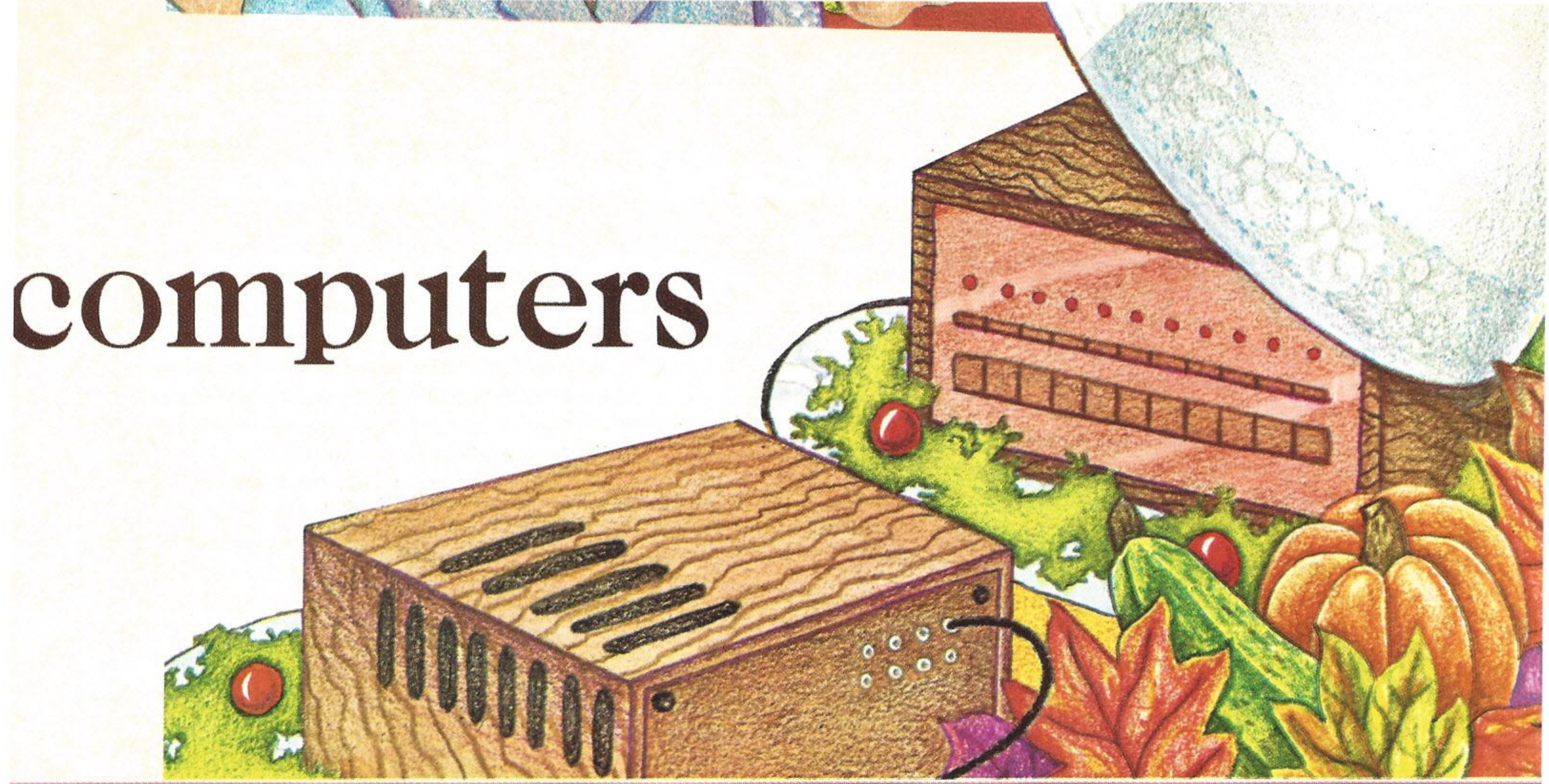
Mainframe microcomputers

The mainframe is essentially a chassis with a central processing unit (CPU) board, built-in power supply, and connectors for plugging in additional printed circuit boards. The mainframe may or may not have a front-panel display. Power and all pertinent microcomputer signals are brought out to a common bus, servicing the printed circuit board plug-in connectors. The advantage of the mainframe is its versatility; by simply adding the proper plug-in boards, the system can be configured in any way desired. For this reason the mainframe will continue to be the best choice for laboratories requiring computer configurations for a variety of applications. In packaging, the mainframes most closely resemble minicomputers and large-scale computers. In the appropriate configuration they are well suited for performing all three instructional functions indicated in "Microcomputers for Science Education" (see box).

You must be very careful, however, when buying a mainframe computer to verify that the configuration selected is adequate to do the job. You could spend a considerable amount on a "system" only to find you need three or four additional boards and an I/O device to make the system work. For ease in trouble shooting and program modification, you may want front panel provisions for examining and changing memory.

Mainframe computers are designed for general purpose use, from machine-level processing to high-level language based computing. If you need one microcomputer to perform all three instructional functions, mainframes prove a well-suited, cost-effective solution.

A recommended configuration for a mainframe-based system for science education and the price range of commercially available units are shown below:



Hardware	Price
Mainframe computer	See Table I
Parallel and Serial I/O (at least one port each)	$150
8K memory, assembled	$250-300
Keyboard (ASCII encoded)	$40-100
Video monitor or modified TV	$150
Video interface board	$150
Cassette Tape recorder	$50
Cassette interface	$100-150

Software	Price
Assembler	0-$100
BASIC	0-$500

Tutorial Systems

Tutorial computers are designed for instruction in machine language programming and, in some cases, hardware design. These systems consist of a single board computer, keypad and display (usually 7-segment LEDs) to facilitate program monitoring. Many units come with a ROM program that monitors a keypad and executes such functions as examining and changing memory, examining and changing the contents of registers, single stepping through the program, monitoring execution through the use of break points, and execution of other useful operations. These features make program entering, debugging and execution readily apparent and are valuable for learning machine language. The first three units in Table I are widely used for sèminars and short courses in microcomputers.

Several of the units were operable as we received them; some, however, required external power supplies. Since the primary purpose of tutorial systems is instruction rather than computation, the determining factors in the selection of such a system are the quality of the overall system documentation and a well-designed student/computer interface. Documentation should include adequate start-up procedures, adequate theory of operation and sample applications programs. The most common weakness in the documentation of the tutorial computers as well as the other three categories of microcomputers was the lack of clearly identified, systematic instructions for checking out, troubleshooting and bringing up the machines. In most cases all the necessary information was hiding somewhere in the instruction manuals, cover sheets, programming manuals or kit building instructions, but in one case we dug for about two hours just to find out how to hook up the power connections. With a few notable exceptions, hardware design sophistication greatly exceeded the sophistication of the accompanying documentation.

The recommended minimum configuration for a tutorial system for science education includes:

Hardware	Price
Tutorial Unit	See Table I
Necessary power supplies	$50

Software	
Keyboard or toggle monitor	with system

"Black Box" complete BASIC Systems.

The "Black Box" differs from the mainframe primarily in its packaging — it is a prepackaged, ready to run, stand-alone system incorporating video driver and cassette interface. The primary appeal of these computers is the high-level language capability they provide for those unfamiliar with or uninterested in hardware. While BASIC is the standard high-level language, other languages such as PL/M FORTRAN, and

Microcomputers for Science Education

Microcomputers can serve three reasonably distinct functions in preparing students for scientific careers. These include instruction in digital logic and machine language programming; instruction in a high-level language such as BASIC; and on-line control and analysis of experiments. When considering the purchase of a microcomputer you should clearly understand its intended purpose, since this will determine which computer and configuration is most suitable.

Instruction in Digital Electronics

At University of Wisconsin-Parkside we've developed a series of experiments in digital electronics which begins with the characteristics of individual logic gates, connects gates to perform combinational logic functions, examines the properties of an arithmetic-logic unit, extends this analysis to the concept of machine language programs and interfaces a microcomputer for processing laboratory signals. A reasonably flexible microcomputer is essential for the last two experiments and useful in demonstrating concepts of binary arithmetic in the first three. A good understanding of digital electronics at this level helps students appreciate "what really goes on" inside a computer running a high level language, and is absolutely essential in designing more advanced experiments requiring computer monitoring and control. Several units are very well designed to serve this objective.

Instruction in High-Level Programming Language

As the ratio of performance to cost of microcomputers improves, some serious questions arise on the cost effectiveness of traditional, relatively expensive time-sharing computer systems for teaching computer programming. Complete microcomputer-based systems include a keyboard for input, video output, cassette tape or floppy disk for mass storage and well-documented software packages including BASIC. Within minutes of unpacking several such systems, we were entering and running BASIC programs. The cost of systems now available ranges between $600 and $1500 and is comparable to the cost of present "dumb" hard-copy terminals in time-sharing systems. If the primary educational function you're considering is instruction in BASIC language programming, we suggest that a system of standalone microcomputers may be more cost-effective than a time-shared minicomputer.

Many of the immediate objections to such microcomputer based systems – relatively slow CPUs, limited high-speed memory, lack of extensive library software packages and absence of hard-copy – are usually invisible or of little consequence to the new programming students. In fact, in one benchmark timing test, we compared the Sol-20 to our minicomputer-based 32 terminal HP2000/ACCESS time-sharing system. Running identical BASIC programs, the minicomputer was about 3 times faster than the microcomputer if it was running on only one terminal. However, when 3 other terminals were running compute-bound jobs, the mini and micro ran at the same speed and with 4 or more terminals the micro beat the mini. In Figure 1 we list the program and present the results of this test. The obvious advantages of the microcomputer – no central computer rental charges, no communication line rental cost, few mechanical parts to fail and generally simple and transparent operation – require serious consideration to be given to this option for providing programming instruction. Additonally, time-shared systems are frequently not operated as machine language programming or on-line data acquisition facilities. Broader institutional considerations will often determine the selection of a particular computer system, but there is no question that competitive pressure from microcomputer systems will improve the cost effectiveness of the service provided by large computer utilities.

Experimental Control and Analysis

As the "human engineering" of microcomputers continues to improve the ease of programming and communicating with analog signals in the outside world, the range of their applications in experimental control and analysis will expand. Present "smart" instruments such as signal generators, oscilloscopes and spectrum analyzers seem to prove the theorem "Any good instrument is even better with a microprocessor." Any experiment which involves several variables, particularly if the variables are available as electrical signals, is a natural candidate for incorporating a microcomputer. The trend in experimental and electronic design is to substitute software for hardware, and the microprocessor represents the "state-of-the-art" in such designs.

In actual experiments, interfacing problems can become quite complex, but the main ideas involved can be clearly demonstrated with several of the units described in the main article in conjunction with inexpensive A/D and D/A integrated circuits. We have developed two exercises in our electronics course at UW-P to demonstrate these techniques. In the first we use a simple lock-up table algorithm to linearize a thermistor to produce a digital thermometer, and in the second we use this thermometer in a feedback loop to construct a heating system which will regulate on a desired temperature profile.

PASCAL are becoming available (but usually at high cost). We tested three such systems and each provided the quick set-up, high-level language capability and ease of operation that makes these models so desirable. These devices are easily adaptable for on-line experimental control and can be used to teach digital logic and machine language. Sophisticated assemblers are available for most units.

The recommended configuration for the "Black Box" system for science education is listed below:

Hardware	Price
Black Box System	See Table I
16K usable RAM memory	$450
Video monitor	$150
Cassette Recorder	$50

Software	Price
Monitor	with system
Assembler	0-$50
High-level language	0-$100

Single Board Computers

The single Board Computer (SBC) may be a naked little utilitarian computer intended for low-level control applications, or an all-in-one-including-firmware-super-board or anything in between. Power supplies and I/Odevices are usually not included, although serial or parallel ports for communication generally are.

The big difference between the SBC and the tutorial microcomputer is that SBCs don't try to teach you all about computing – they assume you already know a good deal. They provide a quick way for someone knowledgeable in digital electronics to get access to microprocessor power at a cost lower than that of a mainframe computer. An SBC is a good choice for jobs requiring dedicated controller or for industrial applications requiring large numbers of identical units. The SBC provides "no-frills" computing capacity for particular applications, but you must provide the necessary application software and I/O peripherals. We would not advise an SBC, however, for those unfamiliar with digital electronics.

The recommended configuration for an SBC system for science education is:

Hardware	Price
Single Board Computer	See Table I
Necessary power supplies	$50
TTY or keyboard and video terminal (the latter can be a do-it-yourself job)	$300-1300

Software	Price
Monitor	with some systems

Evaluation Results

In Table I, we provide information on the characteristics of each of the units for both a minimal cost configuration and a recommended configuration. The prices quoted are current as of spring 1978. The only value judgements in this table involve the choice of the recommended configuration.

Table II presents a summary of our analysis of each of the units tested. Remember we made these value judgements on the basis of using these computers for science education.

Even though several of the units use the same microprocessor chip, each unit has its own unique design, usually through a ROM monitor, which may make it particularly well or poorly suited for the use you may have in mind. For this reason we now look briefly at each unit, highlighting some of the characteristics we believe are worth considering when selecting microcomputers for instructional use.

Tutorial Computers

E&L MMD-1. The E&L MMD-1 is a very well organized, expandable, 8080-based system using as its text the venerated *Bugbooks.* This is the *only* unit we tested which is designed for instruction in the fundamentals of hardware logic as well as programming skills. The E&L series of texts provides an excellent background in digital design. All pertinent signals are brought out to a small breadboard on the top of the board, making experimentation convenient. Manufacturers made a conscious effort in designing the unit to clearly identify the function of various IC blocks on the circuit board.

Data is entered through an octal keypad and displayed in binary code (HI-ADDRESS, LO-ADDRESS, DATA) through 3 groups of 8 discrete LEDs. Operation of the computer is controlled by a Keyboard Executive Program (KEX) stored on a PROM. Space is provided for another PROM so you can store additional user-generated programs.

One negative feature is that the KEX monitor does not include a routine to single step the processor. You could implement this feature using one of the three unused keyboard keys after you become proficient at programming; but from the onset such a single-stepping feature would be valuable. The company does provide an auxiliary module ("outboard option") to perform single stepping, and we recommend it.

The only other design feature you might question is octal encoding on the keypad and displays. We suggest you purchase the hexa decimal data entry-display module. E&L MMD-1 should be considered seriously as a training unit for use in electronics classes. The *Bugbook* lessons can be integrated into the course itself with little modification.

* * *

IASIS-7301 Computer in a Book. The IASIS-7301 comes in a three-ring binder complete with tutorial text, system service manual, programming card and programming scratch pad. With all the necessary information at your fingertips, it's a well-designed system for machine language programming instruction. Data is entered through a 24-key hex keypad and displayed on eight 7-segment displays. Operation of the system is controlled by a keypad monitor which allows single stepping and display of all registers. In addition, the contents of user memory (1K) can be filled from or dumped to cassette tape.

The manual takes you from simple knowledge of binary numbers to fairly complex programming tasks. This unit is an extremely well-engineered system and is also amply documented. We highly recommend the additional application manual (available at extra cost).

This unit is a good choice for a tutorial system, is very easy to use and requires almost no previous knowledge.

KIM-1. KIM-1 is nearly a single board computer, but its ability to work independently of a terminal and the high quality of documentation qualify it among the tutorial products. No lesson-by-lesson instructions come with the KIM-1, but a wealth of information is easily accessible in the well-written manuals that accompany it. KIM-1 provides TTY and cassette interfaces as well as 38 bidirectional I/O lines and an internal timer available to the user. All necessary signals for interfacing are brought out to two 44-finger edge connectors.

As with most tutorial systems, control is provided by a ROM program that monitors the hex keypad and controls the six-digit hex display. Documentation includes the system user's manual, a hardware manual dealing with implementation of the microprocessor and peripheral devices, and the programming manual. Power supplies of +5 volts at about 1.2A and +12 volts at 0.1A (for cassette interface) are required. Two articles in the popular literature describe a TV display interface that you can plug into the KIM-1 applications connector (*Popular Electronics,* July and August 1977. The price in kit form for this board plus a pre-programmed ROM is $40.

KIM-1 makes a good choice for a tutorial computer; it's expandable, well-documented and low-priced. An average knowledge of programming concepts is required.

* * *

INTERCEPT Jr. INTERCEPT Jr. is based on the INTERSIL IM 6100 microprocessor, which emulates the instruction set of the Digital Equipment Corporation PDP-8 machine language. In this respect, it finds no competition and is very cost-effective. Data is entered through an octal keypad and displayed (ADDRESS, DATA) on eight 7-segment LED displays. The keypad monitor on ROM also allows you to enter PDP-8 operation codes.

The PDP-8 instruction set is cleverly designed, and more machine level software has been written for the PDP-8 than for any contemporary mini- or microcomputer on the market. However, the PDP-8 has a unique language and word length, different from other currently-used microcomputer languages. In general we feel the IM6100 is not indicative of the trends of current development, though it may prove the wise choise for particular applications.

Documentation includes the INTERCEPT user manual and an IM6100 data manual.

* * *

COSMAC ELF. One alternative to buying a microcomputer is building your own. A series of articles in *Popular Electronics* (August 1976) describes the construction and operation of an expandable trainer using the RCA CDP 1802. This computer, dubbed "ELF", is a low-cost, labor-intensive way to enter the microcomputer world. The advantages are that if you build from scratch, you can better understand the workings of your computer and thus be better prepared to debug if something goes wrong. The knowledge you gain from such an experience is invaluable.

ELF is now available in two kit forms: A PC board containing a hex keypad and a special chip which does block graphics on a TV monitor; or simply a collection of parts and a reprint of the original article. The ELF is a good, open-ended project in microcomputer design. It assumes a working knowledge of digital electronics.

The basic ELF uses 8 toggles for data input and 2 hexadecimal LED displays. Memory can be examined, changed and write-protected by push buttons and toggles. Memory consists of 256 bytes of RAM.

* * *

INFINITE UC1800. The UC1800 is based on the RCA COSMAC microprocessor. It comes with 256 bytes of RAM. Data is entered and displayed via hex keyboard and a 4-digit (LO-ADDRESS, DATA) hex display. Control of operation is accomplished through a small console of pushbutton and toggle switches which allow you to power on, reset, deposit, execute and single step. Regrettably no provision is made to examine memory without changing its contents.

The UC1800 comes with a printed listing of KEYBUG, a keyboard monitoring system which must be loaded by hand and occupies half the user RAM. It's a fairly long program, and if you make a mistake, you must re-enter from the beginning. This program definitely should have been incorporated into a ROM monitor.

A new kit, the UC1800K2, is available as a special hobbyist package for $159. The UC1800 kit we received should not be assembled by people unfamiliar with electronics. Many of the standard components were not included and the package we received had minimal instructions. In fact, we never did receive a parts layout with the original kit. The manufacturer is now including parts layouts in the instructions for assembly, and the UC1800K2 kit contains *all* parts except cabinet.

Documentation consists of a user's manual, a listing of software available and the RCA microprocessor manual.

Although the price makes the hobbyist kit appear attractive, we do not recommend it unless you have prior kit building experience.

Black Box Computers

Sol-20. The Sol is a versatile terminal/computer system capable of running completely stand-alone or as an intelligent terminal. The basic Sol system comes with keyboard, power supply and main board in a nicely crafted aluminum-and-walnut cabinet. Depending on the package purchased, either 8K or 16K of memory is provided on a board that plugs into the 5-slot motherboard. The motherboard accepts cards compatible with the S100 bus (used by Altair, Imsai, Polymorphic and Vector Graphic).

Sol-20 can communicate with any TTY or RS232 terminal (an EIA standard connector is provided); one parallel 8-bit input and one parallel 8-bit output are provided. The serial interface can run in any of 8 switch-selectable baud rates from 75 to 9600. A cassette interface (300 and 1200 baud) with motor controls capable of handling two cassette recorders at once is provided (one read, one write).

All you need besides the system itself is a cassette tape recorder and a TV monitor or a modified TV set.

System software comes on ROM (SOLOS operating system) and on cassette (5K BASIC, one side 300 baud K.C. standard, the other 1200 baud Proc. Tech. CUTS standard). The SOLOS operating system lets you examine, change memory, execute programs, load from and load to tapes. 5K BASIC comes with the system; 8K BASIC, the ALS-8 assembler and several games (including a particularly addicting version of *Star Trek)* are also available.

One useful function of SOLOS is the ability to stop the display at any time simply by depressing the space bar. This feature helps you keep your sanity while data flies by at breakneck speed. The rate at which characters are printed is also easily set. Several user-defined custom commands are also provided in SOLOS.

System documentation is excellent.

* * *

POLY-88. The POLY-88 system contains an expandable five-slot S100 motherboard, power supply and a cassette interface which can run at 300 or 2400 baud. Included with the unit we evaluated were a 16K dynamic RAM board, keyboard and video monitor, which make the system completely operational with no other hardware required. The POLY-88 also provides an interface for an optional line printer.

System software includes a monitor on ROM which allows examining and changing memory, executing programs, loading from and dumping to cassette tapes and includes a special "front panel" mode which displays memory locations and registers, and allows you to single step while viewing the results. Programs on cassette tape include 11K BASIC (which loads in 90 seconds in the 2400 baud format) and an 8080 macroassembler requiring 8K. Several applications programs and the 11K BASIC package were supplied. Graphics capability, implemented in BASIC, was a particularly strong feature of this system. If you need graphics capability, this computer merits consideration.

* * *

APPLE II. Standard features of the APPLE II personal computer include integer BASIC and Monitor in ROM (8K bytes), color graphics, 4K RAM (expandable to 48K), cassette interface, Apple GAME I/O connector, typewriter-style ASCII keyboard, high-efficiency switching power supply and plastic molded case (4.5" x 18" x 15-1/4"). Additional features include built-in one-pass assembler, disassembler and utility routines and a loud speaker for audible cues and sound synthesis. Floating point 12K BASIC is also available on cassette.

One of the strongest features of the APPLE II is its color graphics capability. By connecting the system with TV monitor or into any standard television using an inexpensive, commercially available RF modulator (not supplied), you can create, with 15 different colors, patterns on the screen with a resolution of 40 horizontal x 48 vertical or 40 horizontal x 40 vertical with 4 lines of text. In a system with at least 12K RAM, high resolution graphics is available with 280 horizontal x 192 vertical or 280 horizontal x 160 vertical with 4 lines of text. The color graphics are easy to use.

This system is well suited for teaching BASIC programming and it has been used for this purpose in a fifth grade class in Kenosha, WI. The color graphics capability gives the APPLE II a great deal of educational potential. One problem with the APPLE II is the sensitivity of the cassette interface to volume control. Unless the volume is set just right, the system will not load programs properly.

Main Frame Computers

Altair 8800B. The Altair 8800B is the third generation of the first, and undoubtedly the most prolific, hobby computer. The basic unit contains power supply, 18-slot motherboard with connectors in 12 slots, CPU and front panel display and control circuitry in an attractive blue cabinet. More boards are available in the S100 bus format (that the Altair introduced) than in any other form. MITS, the manufacturer of the Altair, also offers excellent software packages and, although expensive, MITS extended and DISK BASIC are among the best in the microcomputer industry.

In addition to the basic unit, the manufacturers sent us a 16K dynamic memory board, an RS232 interface board (MITS 2S I/O) and the MITS ACR cassette interface, which runs at 300 baud. With the exception of the cassette interface, all of the evaluation package worked well and seemed to be well designed, sturdily constructed, high quality hardware. The cassette interface board was an exception to the otherwise impressive array of PC boards. It was a concoction consisting of two PC boards held together with stand offs and which had huge discrete components hanging off both sides, precluding the use of the adjacent connector on either side. In addition, it required continual adjustment for satisfactory operation. Subsequent to our evaluation, this board has been replaced by a completely redesigned board.

The Altair system is well documented; and manuals contain photographs, drawings and trouble-shooting aids.

* * *

VECTOR-1. VECTOR-1 consists of power supply and an 18-slot S100 motherboard with connectors in 6 slots. No front panel control is provided, with the exception of the power and reset buttons mounted on an attractive cabinet.

Our evaluation package came with an optional PROM/RAM board. This board contains 1K RAM and has space for 2K x 8 PROM. Ours came with the 256-byte monitor and additional PROM containing Processor Technology's VDM driver program. This board uses a power-on jump to the monitor program.

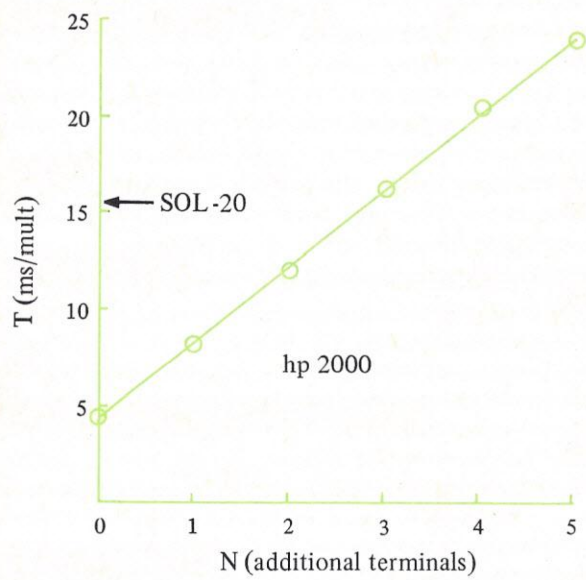
The system documentation was lacking, compared to the well-designed hardware supplied.

* * *

SWTP 6800. Southwest Technical Products' 6800 computer system is based on the Motorola 6800 microprocessor. The chassis contains a seven-connector motherboard and an eight-connector I/O bus. This system is a low-cost alternative to the Altair S100 bus structure with considerable room for expansion. Quite a few plug-in compatible

Benchmark Test

Micro vs Minicomputer




```
TIME

 2  REM:  PROGRAM TO MULTIPLY A
     NUMBER BY ITSELF N TIMES.
 5  N=0
10  X=0
20  Y=X
30  Z=X*Y
40  N=N+1
50  IF N>8000 THEN 70
60  X=N
65  GOTO 20
70  PRINT X,Y,Z
80  END
```

Manufacturers' Comments

To insure accuracy and completeness, the authors showed a preliminary draft of this article to the manufacturers of the computers discussed. Selected excerpts from the manufacturers comments follow:

E&L MMD1. Regarding the negative aspects of non-included single step: To a great extent we've covered this with our new set of operating PROMs that can be obtained as "standard" with the newest version of the MMD-1. The customer has the option of now receiving the unit with a KEX PROM plus an L/D PROM to run the M/I Board; or with a combined KEX-L/D PROM plus our new Monitor PROM that allows single instruction execution of programs. We offer this option because the KEX-L/D operates slightly different from the old KEX and, in some instances, students would be confused by the minor difference. In essense, the second PROM uses up the available space on the main board so there is no direct PROM room available without the add-on M/I Board. We are also supplying full 8 x 512 RAM instead of the original 256 at no extra charge. *– Murray Gallant, president, E & L Instruments, Inc., 61 First Street, Derby, CT 06418.*

IASIS 7301. I find no errors in your comments on the Iasis 7301. However it would be very important to us to have Pasco Scientific, with our address, referenced as a distributor of the Iasis 7301. *– Gene Strohallen, Chief Engineer, Pasco Scientific, 1933 Republic Ave., San Leandor, CA 94577.*

KIM-1. Eric Rehnke of MOS Technology, Inc., 950 Rittenhouse Road, Norristown, PA 19401, responded with some corrections which have been incorporated in Table I and indications that the "Fair" ratings in Table II be changed to "Excellent". *– The Authors*

Intercept Jr. In reviewing the preliminary draft of your article, I offer the following comments/suggestions relating to Intersil's Intercept Jr. unit.

1) The primary intention of the Intercept Jr. unit is not just as an evaluation unit emulating the PDP-8 machine language. Instead, its *versatility* makes it both a 12-bit microcomputer tutorial machine as well as an evaluation unit.

2) The sentence "the IM6100 is not indicative of the trends of current development, . . . " is an unfair statement because, as you may know, several of the current machines use a minicomputer architecture like IM6100. I do not agree with that statement.

3) Intercept Jr. is the only tutorial/evaluation kit which comes ready to use with a set of four batteries, and the kit includes batteries. This is a major feature in that the user does not have to purchase an expensive power supply to use with the kit. This fact is not in your article. *– D. W. Sohn, Microcomputer Marketing Manager, Semiconductor Division, Intersil, Inc., 10710 No. Tantau Ave., Cupertino, CA 95014.*

Infinite UC1800. Bill Haberken, Infinite, Inc., 1924 Waverly Place, Melbourne, FL 32901, responded with corrections which have been incorporated into the text and the following comments:

In regard to the statement "Regrettably, *no* provision is made to examine memory without changing its contents," he responds "Don't understand this comment. There is a single step function. Our application note for expansion will also describe a simple modification for a second type of single step which allows examination without getting involved in program loops."

In regard to the statement. "The kit is poorly documented and the instructions are vague," he responds "Interesting. Other evaluators (and there have been a number) have termed our documentation superior." *– The Authors*

Sol-20. Our new Extended Cassette BASIC is considerably more powerful than the BASIC/5 included with the Sol Systems II and signifies a major step in connection with software support for microcomputers. *– Ralph I. Palsson, Customer Applications Manager, Processor Technology Corp., 7100 Johnson Industrial Drive, Pleasanton, CA 94566.*

Apple II. Phil Roybal of Apple Computer Inc., 10260 Bandley Drive, Cupertino, CA 95014, responded with some factual additions which have been incorporated in the text plus the following comment on the sensitivity of the cassette interface to volume control: "This is primarily a function of tape head alignment. A properly aligned tape deck will allow volume variations of up to 40% without undue problems". He also indicates that the "Fair" in Table II should be a "Good". *– The Authors.*

Super Jolt. Our new single board product, VIM-I, is an ideal unit for educational/training applications. We have already delivered several orders for VIM to be used in just such an environment. *–M.R. Lemas, President, Synertek Systems Corporation, 3001 Stender Way, Santa Clara, CA 95051.*

Technico-SSB. In the description of our board, the IIA which allows programming in assembly language for only $49 is not mentioned. I feel this is, along with the E-PROM programmer, a major feature of our board since it allows assembly language programming at an extremely low cost.The other feature of our board is that it is not a "throw away" item but can be used as the CPU of our fully expanded box system. I believe we provide the only service in the industry whereby a kit can be returned for a flat $25 fee and returned to the user in working order. Relative to your charts, you list that we have available no mass storage for the Technico board. We have been shipping, for some time now, a 4800 Baud, 30,000 Byte digital cassette which plugs into the CRU jack on the starter board. *– Bill Regan, President, Technico, Inc., 9130 Red Branch Rd., Columbia, MD 21045.*

TI 990/100M. John Caulfield, Texas Instruments, Houston, TX, called with some factual corrections which have been incorporated into the final draft. He also indicated that a new T.I. microcomputer, the TMS 9980, would soon be available at $195 for use in educational environments. *– The Authors.*

boards are available for this system, and included with the basic unit are 4K of RAM and a TTY and RS232 interface capable of running at 110 and 300 baud.

There are no front panel lights or switches, but with the addition of a serial terminal you can immediately begin execution of the MIKBUG monitor program (stored in ROM) by simply turning on the power and hitting reset. This program lets you examine and change memory, examine the registers, begin execution of user programs, punch and load paper tape and service intercepts.

Documentation is extensive, although very spotty and disorganized.

Single Board Computers

SUPER JOLT. The CP 110 "Super Jolt" is a monster of programming power packed into a minature 5" x 7" board. Primarily intended for OEM use, the Super Jolt provides 28 bidirectional I/O lines, interval timer, TTY and RS232 interfaces and a ready-to-run DEMON debug monitor. 1024 bytes of RAM are provided for the user.

Three voltages are required – +5 volts at 750 mA, +12 volts at 200 mA and -10 volts at 20 mA, (required for TTY interface). When you power up and reset the board (which must be done by grounding a pin on a connector), the monitor responds by displaying the processor status and registers. The DEMON monitor, a well-designed programming aid, simplifies system operation. The Super Jolt can talk to serial terminals at rates up to 300 baud.

For an extra $200, you can purchase 4K of ROM, which plugs into two empty sockets on the Super Jolt board. This ROM contains Tiny BASIC and a resident assembler program, giving this board impressive performance for its size. Our evaluation model came with these additional ROMs.

Documentation includes the DEMON user's manual and manuals for the assembler and Tiny BASIC programs. All were adequate for people with a knowledge of 6502 language to begin programming, but no programming manual was provided. This well-designed board offers good performance at a relatively high cost.

* * *

TI 990/100M. The TI 990/100 computer board falls between a large micro and a small minicomputer. It's designed around the TMS 9900, a 16-bit microprocessor with the same instruction set as Texas Instrument's 990 line of minicomputers. The 100M board comes with 20 mA current loop interface, RS232 interface (on an EIA standard connector), 256 words (512 bytes) of RAM (expandable to 512 words) and the TIBUG monitor program on EPROMs. The unit is completely compatible with the TI 745 terminal. In general, programs written for this machine will run on the entire TI 990 computer line.

Documentation is excellent. The TIBUG monitor features hexadecimal arithmetic functions, breakpoint and single step modes of operation. This is a powerful nucleus for many applications; its main strength is in the minicomputer-like architecture of the TMS 9900 chip.

* * *

Technico Super Starter Board (SSB). The SSB from Technico is also based on the TMS 9900 16-bit microprocessor from Texas Instruments. The board includes TTY and RS232 interfaces which can run at rates from 110 to 9600 baud, 256 words of user RAM (expandable to 1K words) and space for two 2708 PROMs which the user may program using the onboard programmer and writing into the PROMs. Power requirements are +12 volts, +5 volts at 1 A, -5 volts at 250 mA and +28 volts for PROM programming. All power and I/O connections are brought out to 16-pin dip IC sockets.

A good monitor, provided on ROM, features breakpoints, hex arithmetic, "snaps", loading and dumping operations. System documentation is fairly complete, although our manual contained several empty sections. Technico SSB is a low-cost alternative to TI's SBC, and is designed to provide minicomputer power in a small system.

Kit Evaluation

All of the kits we receive did work assembled, although in some instances, if we hadn't detected errors in the kit instructions, we might have assembled the units incorrectly. No kit manufacturers provided adequate trouble shooting procedures, in our judgement, for the average kit builder.

We summarize here some of the special features of the kits we constructed and the problems encountered.

A Note on Kit Building

Building a microcomputer from a kit can be a frustrating experience. You must realize that even if all directions are followed exactly (or even better, if the kit is constructed correctly), there remains a chance it will not work. Perhaps not all integrated circuits were tested before shipment; ICs usually fail, if they ever do, during the first few hours under power. A *good* technician may take a day or two to find the problem. So, unless you have access to a good oscilloscope and a reasonable supply of spare parts, trouble-shooting may prove impossible.

When you build a kit, you get no guarantee of what will happen when the power goes on. But we're not saying that kit building is an unfavorable risk or an unwise decision. Most manufacturers will trouble shoot an ailing computer and repair it at a nominal cost.

The kit-versus-assembled-unit decision also involves the intended purpose of the unit and experience of the builder. If money permits and you don't have considerable experience in digital electronics but need a unit guaranteed to work, an assembled unit is your only option. If, on the other hand, you have the instruments, time and knowledge required for building and debugging kits, you can obtain a good microcomputer at very low cost. In fact, by using component lists and printed circuit boards which are available for a number of these units, you possibly can avoid capital equipment limits and build microcomputer systems from your supply budget alone. For example, complete microcomputer chip sets featuring "cosmetic reject" 8080A and 8085 microprocessors are available from the INTEL Corporation for $20 and $40 respectively. Finally, you shouldn't overlook the educational value for both instructor and students of constructing the computer from parts – even though the main lesson may be that "Life is not always easy".

A final source for low-cost microcomputers should be mentioned – electronic supply houses and kit manufacturers. Some very interesting units, not available during the evaluation, are now available (for example a Heathkit 8080A based kit for $375 and a Radio Shack Z80-based assembled and tested system with BASIC for $599.95).

E&L MMD-1. The E&L assembly instructions are generally very clear, complete, well ordered and easy to follow. We found one minor error; no mention was made of a connector between the power lines on the PC board and the four color coded binding posts.

* * *

INFINITE UC1800 (Hobbyist). This kit comes without a transformer, most of the sockets, switches and ordinary discrete components to keep the "hobbyist" kit cost low. This kit is poorly documented and the instructions are vague. The new UC1800 K2 kit does contain all parts.

* * *

VECTOR-1. Instructions are presented in generally the right sequence, although the text is sometimes wordy and confusing. The kit is organized as a number of small kits, each with its own separate manual which rarely relates to any of the others. No start-up instructions are provided for the computer as a whole, and you must look around on the bench to see if all of the parts are gone to determine that all is done. No mention was made of a connection between a wire from the power supply to the lamps in the power and reset buttons.

* * *

Technico SSB. The instructions were generally sufficient but we found some confusing typographical errors. The instructions were well organized and kit construction was simple.

* * *

Sol-20. Sol-20's assembly instructions were the most complete of any unit supplied. The instructions contain several intermediate tests which allow each functional block of the computer to be tested independently. We found two problems. A substitution was made for one of the transistors used in the keyboard assembly, and the new transistor had a different case type than the original. Also, the +5 supply was a bit temperamental and did not regulate correctly at first.

* * *

SWTP 6800. The assembly instructions are fairly good. The boards are not tightly packed (low part density), thus aiding assembly. Cursory checkout procedures are provided.

Conclusion

There are many excellent microcomputers already on the market and, in the proper configuration, they may serve very effectively as instructional tools. To use them most effectively, however, you must have the intended instructional purpose clearly in mind and select the microcomputer configuration adequate for the task.

If your intended purpose is instruction in digital logic and machine language programming, we recommend any of the first three "Tutorial" computers listed in Table I (E&L MMD-1, IASIS 7301, and KIM-1). If the emphasis is on hardware design, the E&L MMD-1 has the edge since it is the only computer designed specifically with this purpose in mind. If your emphasis is on machine language programming, the IASIS 7301 offers a slight advantage because of its excellent documentation and convenient monitor program with single step and register interrogation features.

If instruction in a high-level language such as BASIC is the intended function for the microcomputer, we recommend any of the three "Black Box" computers in Table I (Sol-20, POLY-88, and Apple II). Because of the sophisticated BASIC software available and the ease of setting up and operating these systems, we believe they should be seriously considered as an alternative to the time-shared minicomputer-based systems, particularly for introductory level instruction where large system libraries may not be essential. Keep in mind that these microcomputers can also perform the functions of machine language programming instruction and laboratory experimental control, both of which may not be feasible on most time-sharing systems because of their "dedicated" nature.

Finally, if your objective is on-line control and analysis of laboratory experiments, the single board computers listed in Table I may be the most cost-effective solution, particularly if the hardware and software development costs can be prorated over several identical systems. Such applications development also requires an expert understanding of microcomputers and digital electronics.

The main argument in favor of the "mainframe" computers listed in Table I is that they can perform well all three instructional functions and others as yet unforeseen. This flexibility makes them powerful tools for applications development. We have used several Altair 8800s in a variety of applications, and do not hesitate to recommend these systems as fine, general purpose microcomputers.

As a final observation, we note that the only way to get into the microcomputer field is to obtain a unit, sit down with it and make it work. As you gain experience, a whole new world of applications opens up. The only limitations are your own imagination and the time you can spare for development. □

Microprocessor Characteristics

As the "Microprocessor" column in Table I indicates, all of the units tested are based on one of the following microprocessor integrated circuits: INTEL 8080A, RCA COSMAC CDP 1802, MOS Technology MCS 6502, Motorola 6800, INTERSIL IM6100, and Texas Instruments TMS 9900. These ICs span the spectrum in price and sophistication from the inexpensive "standard of the industry" 8080 to the powerful TMS 9900 (really a minicomputer trying to enter the microcomputer market).

Two of the most important factors determining the "computing power" of a microcomputer chip are the number and sophistication of the machine language instructions and the average instruction time. Since instruction sets and times vary widely, you must evaluate them according to your particular needs. To estimate how these chip characteristics affect performance, we wrote machine language programs to multiply two numbers stored in memory and to store the double precision product in memory 65,000 times. We feel this benchmark gives a reasonable balance of logic, arithmetic and transfer-of-control instructions. The benchmark results are given in Table III.

We must emphasize, however, that the "computing power" implied in Table III may be the *least* important consideration in selecting a microcomputer for instructional purposes. Much more important criteria include quality of documentation, ease of communicating with the computer, software available and number of units on the market and in the field.

Table I Microcomputer Characteristics

Microcomputer	Category	I/O Devices	RAM Memory	Mass Storage	Auxillary Power Supply (Volts)	Micro-Processor	Clock Freq. (MHZ)	Software Supplied	Base Price (Kit/Assembled)	Price of Recommended Configuration (Kit/Assembled)
E & L MMD-1	Tutorial	Keypad, Binary LEDs 1 I/O Port	8 x 256	None	None	8080A	0.750	Keypad Monitor	$423/600	$644/890
IASIS 7301	Tutorial	Keypad, Hex LEDs 1 I/O Port	8 x 1024	Cassette Interface	+5,+12	8080A	2.	Keypad Monitor	----/450	----/458 & P.S.
KIM-1	Tutorial	Keypad, Hex LEDs Serial TTY, 15 I/O Lines	8 x 1024	Cassette Interface	+5,+12	MCS 6502	1.	Keypad Monitor, TTY Monitor	----/245	----/245 & P.S.
INTERCEPT, Jr.	Tutorial	Keypad, Octal LEDs	12 x 256	None	4 D-Cells	IM 6100	2.5	Keypad Monitor	----/281	----/426
COSMAC ELF	Tutorial	Toggles, Hex LEDs	8 x 256	None	+5	CDP 1802	≤2.	"Mini Monitor"	80-100/---	80-100/--- & P.S.
INFINITE UC1800 (Hobbyist)	Tutorial	Keypad, Hex LEDs	8 x 256	None	None	CDP 1802	2.	Keypad Monitor	130/389	130/389
SOL-20	Black Box	Keyboard, Video Display, TTY, RS232, Printer Int.	8 x 2K	Cassette Interface	None	8080A	2.	5K-Basic, Monitor, Video Driver	575/745 (2k kit)	1883/2283 (16K)
POLY-88	Black Box	Keyboard, Video Display, Printer Int.	8 x 16K	Cassette Interface	None	8080A	2.	11K-Basic, Monitor, Video Driver	1500/2500	1500/2500
APPLE-2	Black Box	Keyboard & Color Video Display (Graphics)	8 x 4K	Cassette Interface	None	MCS 6502	1.	12K-Basic, "Disassembler"	----/995	----/1445 & Color Monitor
ALTAIR 8800B	Mainframe	Front Panel Toggles binary LEDs	None	None	None	8080A	2.	None	750/995	1290/1695 & Software
VECTOR-1	Mainframe	None	None	None	None	8080A	2.	None	500/690	1040/1390
SWTP 6800	Mainframe	TTY, RS232 Interface	8 x 4K	None	None	6800	1.	Monitor	430/---	970/---
SUPER JOLT	Single Board	TTY, RS232 Interface, 28 Parallel I/O Lines	8 x 1024	None	+5,+12, -10	MCS 6502	1.	Monitor	----/375	-/675-1675 & P.S.
TECHNICO-SSB	Single Board	TTY, PS232 Interface	16 x 256	None	±5,+12	TI 9900	2.	Monitor	299/399	600-1600/700-1700/& P.S.
TI 990/100M	Single Board	TTY, RS232 Interface	16 x 256	Cassette Interface	±5,+12	TI 9900	2.	Monitor	----/450	-/750-1750 & P.S.

Table II

Evaluation Results

Microcomputer	Hardware Expandability	Software Availability, Recommended System	Ease of Interfacing	System Documentation	Background Prerequisite (Kit)	Background Prerequisite (Assembled)
E & L MMD-1	Good	Fair	Good	Excellent	Minimal	Average
IASIS 7301	Fair	Fair	Fair	Excellent	--	None
KIM-1	Good	Fair	Fair	Excellent	--	Average
INTERCEPT Jr.	Good	Fair	Fair	Good	--	Average
COSMAC ELF	Poor-Fair	Fair	Poor-Good	Good	Expert	--
INFINITE UC1800 (Hobbyist)	Fair	Fair	Poor	Poor	Expert	Expert
SOL-20	Excellent	Excellent	Excellent	Excellent	Average	None
POLY-88	Excellent	Excellent	Excellent	Excellent	--	None
APPLE-2	Good	Excellent	Fair	Good	--	None
ALTAIR 8800B	Excellent	Excellent	Excellent	Good	Average	Average
VECTOR-1	Excellent	Excellent	Excellent	Poor	Average	Average
SWTP 6800	Excellent	Excellent	Excellent	Good	Average	--
SUPER JOLT	Good	Good	Good	Good	--	Expert
TECHNICO-SSB	Good	Fair	Good	Good	Average	Expert
TI 990/100M	Good	Fair	Good	Excellent	--	Expert

Rating Scale

Hardware Expandability:

Excellent: Many direct plug-in products available; built-in bus structure.

Good: Some products available, connectors supplied.

Fair: No products available, signals available.

Poor: No products available, no signals available.

Software Availability for Recommended Configuration:

Excellent: High-level Language, Assembler, Operating Sys.

Good: Minimal BASIC, Monitor

Fair: Minimal Monitor

Ease of Interfacing:

Excellent: Parallel and serial I/O ports available, in addition to communication ports, no additional wiring required.

Good: Latched ports available, additional hardware required.

Fair: Minimal I/O capability

Poor: No general purposes I/O ports provided

Background Prerequisite (Kit):

Minimal: Some kit building experience

Average: Experienced kit builder

Expert: Considerable experience in kit building and digital electronics

Background Prerequisite (Assembled):

None: No previous computer experience

Average: Familiarity with programming concepts

Expert: Experience with machine and assembly language programming.

Table III

Microprocessor Characteristics and Benchmark Results

Microprocessor	Word Length (bits)	Max Addr. Memory (words)	Semiconductor Type	P.S. (volts) required	Add Instruction Time/word (μs)	#Machine Lang. Instr.	Clock Rate (Mhz)	#Words req. for Mult. Program	Time for 65,000 Multiplies (seconds)
INTEL 8080A	8	65K	NMOS	+5,-5,+12	2.0	78	2.0	87	49.0
RCA COSMAC CDP 1802	8	65K	CMOS	3 to 10V	5.0	91	6.0	68	38.0
MOS TECHNOLOGY MCS 6502	8	65K	NMOS	+5	1.0	139	2.0	75	13.0
MOTOROLA 6800	8	65K	NMOS	+5	2.0	72	1.0	52	21.0
INTERSIL IM6100	12	32K	CMOS	+4 to +10V	2.5	70	8.0	54	70.0
TEXAS INSTRUMENTS TMS 9900	16	32K	NMOS	+5, ±12	7.3	69	3.0	13	2.5

How to Write for Personal Computing

Have you programmed your computer to converse in Gaelic? to do your home-ec homework? to read a bedtime story to the kids? Are you a frustrated fiction writer who's caught the computer bug? Or, have you found the ideal system or the absolutely worst combination of components?

Why not share your experiences with our readers? Yes, you too can write for *Personal Computing.* You choose the topic, *any* topic. If your topic relates to computers, great. If it relates to personal computers, even better. Computer hobbyists are looking for an excuse, any excuse, to buy a computer, and you might just offer the justification they're looking for.

We accept articles for all our sections – *Launching Pad* (our tutorial section for beginners), *On the Lighter Side* (where we print humorous applications), *In the Money* (how to use your computer to benefit financially), *Digging In* (for our more "advanced" topics), and *Once Upon a Time* (where we let your imagination run wild). We'd love to see some comparisons of computers or computer products. Tell us the good *and* bad of your system.

Keep your writing simple. No, our readers are not simpletons or beginners, but if you can explain something in simple words, do so. Don't clutter your piece with unnecessary jargon. If you're already into computers, give the newcomers a hand and let them in on some of the tricks of the trade – in simple terms. Examples, analogies, and charts and diagrams help both the beginner and the more advanced user appreciate what you're saying. Feel free to use "I" and "you" to make your article more personal and meaningful to the reader. Put the reader in the position of programmer ("you"). Also, please do not write your entire article in caps. And please indent for each paragraph.

Some things to note. Make sure your details are accurate – especially prices, other numerical information, and company names. Don't rely on hearsay or memory.

If you write about a program you've invented, try this order (to make sure you cover all angles): state the program's purpose; show a sample run; explain what the input options are, and what the output means; show another sample run; explain the underlying theory (if any); state the language, version, and computer you used and their peculiarities; show the listing; explain the program's over-all structure; analyze the program's details line by line; and suggest how the reader might improve or change the program.

Whatever your area of interest, you can turn it into an article. For example, if you're interested in watching birds then why not try an article on how to use a computer to track bird migrations? Or if your business is ____________________, why not try a piece on computers and how they can be used to ______ ________________. We're open to ideas . . . .

If you've never written for a publication before and you'd like to discuss your piece with us before beginning it, give us a call. (Please do *not* mail us vague story proposals or outlines. We'd rather see the first few paragraphs of your article.) We'd be glad to discuss what you have in mind, and offer a few ideas of our own

As a matter of form, we prefer (and are more likely to accept) articles that have been typed. Most of our articles run around 2-4 magazine pages. (There's about 3-3/4 typewritten pages to a magazine page.)

Now here's the good part: we pay for any original material we print, although the price varies depending on the *quality* of the article. (So make it good!)

Why not give it a whirl? There may be a latent Hemingway, Fitzgerald or Asimov beneath that Einsteinian exterior.

Sample Run I

```
MESSAGE:

BUDGET $560 FOR CHRISTMAS

NUMBER OF DAY  ? 0

ENTER A ZERO FOR THE NUMBER OF THE MONTH WHEN YOU HAVE FINISHED.

WHAT MONTH NUMBER? 0

                    SUSPENSE FILE ROUTINE

SELECT: PRINT (1); ADD (2); DELETE (3); OR END ROUTINE (4)? 3

                    DELETE FROM THE SUSPENSE FILE ROUTINE

ENTER A 'ZERO' FOR THE NUMBER OF THE MONTH WHEN FINISHED.

NUMBER OF THE MONTH  ? 2

EACH ENTRY FOR THE MONTH WILL BE PRESENTED TOGETHER WITH THE
DELETE OPTION.

 20 THIS IS TEST 4 FOR DAY 20 FOR MONTH 2
DELETE?

 10 THIS IS TEST 5 FOR DAY 10 FOR MONTH 2
DELETE?

 30 THIS IS TEST 6 FOR DAY 30 FOR MONTH 2
DELETE? YES

THE ITEM HAS BEEN DELETED.

ENTER A 'ZERO' FOR THE NUMBER OF THE MONTH WHEN FINISHED.

NUMBER OF THE MONTH  ? 0

                    SUSPENSE FILE ROUTINE

SELECT: PRINT (1); ADD (2); DELETE (3); OR END ROUTINE (4)? 1

                    PRINT SUSPENSE FILE ROUTINE

OK
RUN

                    SUSPENSE FILE ROUTINE

SELECT: PRINT (1); ADD (2); DELETE (3); OR END ROUTINE (4)? 2

                    ADD TO THE SUSPENSE FILE ROUTINE

ENTER A ZERO FOR THE NUMBER OF THE MONTH WHEN YOU
     HAVE FINISHED.

WHAT MONTH NUMBER? 6

YOU ARE ALLOWED A MAXIMUM OF 10 MESSAGES PER MONTH.

ENTER A ZERO, COMMA, ZERO WHEN YOU HAVE FINISHED
     FOR THE MONTH.

NUMBER OF DAY  ? 15

MESSAGE:

BUDGET 1250 FOR VACATION.

NUMBER OF DAY  ? 10

MESSAGE:

AUTO INSURANCE IS DUE $542.34

NUMBER OF DAY  ? 0,0
?EXTRA IGNORED

ENTER A ZERO FOR THE NUMBER OF THE MONTH WHEN YOU
     HAVE FINISHED.

WHAT MONTH NUMBER? 9

YOU ARE ALLOWED A MAXIMUM OF 10 MESSAGES PER MONTH.

ENTER A ZERO WHEN YOU HAVE FINISHED FOR THE MONTH.

NUMBER OF DAY  ? 1

MESSAGE:

ANTICIPATE 7% COST OF LIVING RAISE IN PAY.

NUMBER OF DAY  ? 0

ENTER A ZERO FOR THE NUMBER OF THE MONTH WHEN YOU
     HAVE FINISHED.

WHAT MONTH NUMBER? 12

YOU ARE ALLOWED A MAXIMUM OF 10 MESSAGES PER MONTH.

ENTER A ZERO, COMMA, ZERO WHEN YOU HAVE FINISHED
     FOR THE MONTH.

NUMBER OF DAY  ? 1
```

Continued from Page 55

of the sub-file, as where that value is zero. Statements 980 through 1100 form a chronological sort routine, sorting on the day field and carrying the remainder of the record with it.

In statement 990 the "swap flag", LF, is initialized to zero; in statement 1070 it is set to one if a swap occurs; and in 1090 the flag is tested to see if a swap occurred. If not, the swap routine is exited.

Table I – Program 4 Changes

```
THE CHANGES INDICATED BELOW (UNDERSCORED PORTIONS) MUST BE MADE TO
THE TRANSACTION ENTRY PROGRAM

CRAS-TE  (PROGRAM 4)
LIST320-420

320      IF M=M1 OR M=M1+1 THEN 395
330      IF M>M1+1 THEN PRINT "YOU ARE SKIPPING A MONTH.  PLEASE REENTER THE
            MONTH NUMBER.": PRINT: GOTO 300
340 IF M => M1 THEN 395
350      PRINT "TRANSACTIONS FOR THE MONTH OF " M$(M)" HAVE BEEN CLOSED OUT"
360 PRINT "BY TRANSACTIONS ENTERED IN " M$(M1)".   YOU MAY ENTER YOUR NEW"
370 PRINT "TRANSACTIONS IN " M$(M1) " OR LATER.": PRINT
380 PRINT "PLEASE REENTER THE MONTH.": PRINT
390 GOTO 300
395 IF M>M1 THEN 412
400 M=M+1
410 GOSUB 1680: M=M-1: GOTO 420'            * TO GET OPENING BALANCES
411 '
412 GOSUB 1680'                             * GET CLOSING BALANCES OF PREVIOUS MONTH
414 M=M+1
416 GOSUB 1890'                             * FILE OPENING BALANCES FOR NEW MONTH
418 M=M-1
419 '
420 Q=Q1: QN=QM
```

Notice, too, that each cycling of the loop from 990 to 1100 brings the record number with the lowest date to the top of the subfile. It would be inefficient to allow the loop to cycle to the top each time. Thus, the highest value to be incremented in the loop, "J4", is reduced by one each time the outer loop is cycled, as in 990.

Statement 1190 ensures the page contains room for a full month's messages; otherwise the printer advances to "top-of-form". Page numbers are incremented in statement 1220.

The loop beginning with statement 1750 and ending with 1840 presents each suspense file entry for the month for review and asks if you want to delete it. A carriage return will suffice where the entry is *not* to be deleted. Otherwise, a "Y" or "YES" will delete the item.

Statement 2050 is a standardized routine for centering and then printing variable length titles on a page. The generalized variable to be printed is "Y$". Thus the body of the program concatenates and/or translates

the string where the string is known by a different variable name. See, for example, statements 610 and 630.

Sample Run 1 represents messages appearing on the CRT during a program run. The Suspense File Menu appeared with 4 choices. The program loops back from whichever of the first three options are selected until option 4 is elected. At this point the program calls the CRAS-RS (Routine Select) program to run.

In this example, a file had already been created, and the user wanted to add to it. After making a number of entries, he indicated he was through by escaping the month loop by entering a zero as the date. Then he escaped the routine loop by entering a zero for the month.

This entry brought the user back to the menu, and a DELETE routine (3) was selected. Each message of the month(s) was reviewed and the user indicated whether to delete the message.

Finally, the user elected option "1", PRINT, and the entire file was printed in format (Sample Run 2).

CRAS-CP (Program 11)

The Check Printer program presented the most interesting challenge because of the requirement of converting numbers to word strings, together with the other string material which makes up conventionally expressed written values. I believe there must be an easier method than the one I developed; I welcome readers' comments on improved methods.

During the Transaction Entry Program, you indicate whether you want to print a check. If you want a check, a string field in the transaction entry record is filled with stars (****). These stars will be replaced by the check number at a later time.

The program assumes transaction entries may be made daily or less often, and yet checks prepared only at designated intervals. Should the Check Register be printed during one of those intervals, the entry in the check number column will in fact show four stars, thus indicating a check not yet issued. For checks already issued, the check number will replace the stars.

Prompts appears on the CRT when you run this program. The first prompt asks for the first transaction number for which checks are to be printed. You need not be accurate in your response so long as you select an earlier transaction number than that of the first transaction for which a check is yet to be printed. If you don't, the computer shifts to the CRT port and announces that CHECK NUMBER ( ) HAS ALREADY BEEN PRINTED ON TRANSACTION ( ), substituting the correct check number and transaction number in each instance.

The second prompt asks for the check number of the first check to be printed. The program assumes preprinted check numbers are used on the checks. It assumes, too, that some checks will be destroyed, thus breaking the series. A final assumption is that the check sequence for a particular run will be an unbroken series. Each time a check is printed, the check number will increment by 1, and the new value is printed on the next transaction entry record to which it relates.

Notice the caution printed in lines 740 and 750. Although the print statements are formatted for conventional check, you may need to adjust the tabs and indexes in the print loop from 770 to 950.

The most interesting feature of the Check Printer Program is the numbers-to-string-words conversion, the program begins in this conversion by reading the data file, 1090 to 1180. The

Sample Run II

```
================================================================================
                          SUSPENSE FILE FOR 1978

      JOHN AND JANE DOE (1234-56-78) / FIRST NATIONAL BANK OF ANYTOWN

--------------------------------------------------------------------------------
JANUARY
-------

 1   THIS IS TEST 1 FOR DAY 1 FOR MONTH 1
 2   THIS IS A TEST FOR SUBSEQUENT ADDITION OF MESSAGES.

FEBRUARY
--------

10   THIS IS TEST 5 FOR DAY 10 FOR MONTH 2
20   THIS IS TEST 4 FOR DAY 20 FOR MONTH 2

MARCH
-----


APRIL
-----


MAY
---


JUNE
----

10   AUTO INSURANCE IS DUE $542.34
15   BUDGET 1250 FOR VACATION.

JULY
----


AUGUST
------


SEPTEMBER
---------

 1   ANTICIPATE 7% COST OF LIVING RAISE IN PAY.

OCTOBER
-------


NOVEMBER
--------


DECEMBER
--------

 1   BUDGET $560 FOR CHRISTMAS

--------------------------------------------------------------------------------
```

CRAS System Modifications

Changes, usually minor, are made in the CRAS System from time to time to make it more convenient to use. Following are two such changes which you may wish to incorporate in your system.

Change No. 1: CRAS-TE (Program No. 4, Transaction Entry Routine)

920 AE(R)=EN*A1/ED

Change to read:

920 AE(R)=EN*(A1-AT)/ED

This change lets you apply proportional allocation to the *balance* of a transaction entry amount. Formerly, proportional allocation could only be used with the *total* amount of the transaction entry. For example, take a transaction entry which relates to a home mortgage payment in which the home is also used for business. With this change, you can now enter the amount to be charged off to principal in the ACCOUNT, COMMA, AMOUNT loop, then enter a 0,0 (zero, comma, zero) to enter the proportional allocation loop. At this point, you enter the amount to be charged to the business interest expense account on a fractional basis, and the balance is charged to the personal home mortgage interest account.

Change No. 2: CRAS-CP (Program No. 11, Check Printer)

840 KJ=4: GOSUB 1820'

Change to read:

840 PRINT: PRINT "TXN. #" J: PRINT: PRINT

"KJ" is an index variable in the program which, together with the subroutine at 1820 merely caused the printer to index 4 times. The change permits printing the transaction number on the check. You'll find it more convenient to reference checks to the Check Register by transaction number than by check number. You'll also find this change convenient when clearing checks in the Checking Account Reconciliation Program (CRAS-CAR, Program No. 9).

data is arranged so it will be read into a matrix, thus preserving the numerical relationship of the words – for example, FIVE, FIFTEEN, FIFTY appear from left to right in the sixth row of the matrix.

Statement 720 causes the numerical word matrix to be read (1090 to 1180), after which the check printing loop is entered. This loop indexes from P to Q1. P is the first transaction number to be examined for the Check Print Flag, and Q1 is the last transaction number.

A record is read and the balance of the loop skipped if the Check Print Flag is not set. On the other hand, if a star is read in the K6$ field, the CHECK ALREADY PRINTED prompt appears on the CRT.

To print a check, the printer indexes to the date line of the check (840) and prints it (M$(M)) plus the last two digits of the year. The printer then indexes to the PAYEE line and prints A5$; then tabs to the AMOUNT entry where A1 is printed in WC$ format (star-filled).

Now the program enters the most interesting part – the conversion of digits to word strings. A1, the AMOUNT, is converted to ID, an integer. Then IC, also an integer, is defined as the difference between A1 and ID, multiplied by 100. This yields the cents value as an integer. The rest of line 1230 compensates for floating point "pickups". For example, $14.92 could become $14.9199999. Truncating to an integer without compensating results in $14.91. These inaccuracies are inherent in the interpreter's handling of floating point.

ID (the AMOUNT) is then converted into a string, ID$, in line 1240. The cents (IC) are dealt with first. If IC equals zero, then a string "AND NO/100" is formed for later concatenation with the dollar amount (ID$). If not equal to zero, the string "AND" plus the string value of IC plus "/100" is concatenated as IC$. Again, this IC$ string will be concatenated with the ID$ string later on.

ID, the truncated AMOUNT, is then converted to a string, ID$, in line (1240), and the length (L) of the string determined in line 1280. An additional character field is picked up in the conversion of a number to a string. A two-digit number will have a string length of three, and so on. Thus, if L exceeds a value of 5, the AMOUNT must consist of more than 4 digits. The check printer is limited to the amount $9,999.99 or less – thus the length-of-string test in line 1290. Where a greater amount is to be stated on the check, all entries will be printed except the amount in words (ID$+IC$). A prompt to this effect appears on the CRT (lines 1300 and 1310).

The program now develops the dollar string, ID$. Any nulls preceding digits contained in the string are discarded in lines 1350 to 1370. Line 1390 then provides the real ID$ and its true number of digits (L). Lines 1410 to 1440 re-convert each digit in turn, from left to right, to a single precision number, thus creating an I-array of L-length. Each I(K) in the array points to the correct row of the number-conversion matrix, A$(J,K) (1090-1180). Now the program must establish the desired column of the matrix for concatenating the ID$ string.

Line 1460 sends the program to one of four subroutines depending on the length (L) of the string, ID$. Where L=1 (the simplest case), the single digit dollar subroutine is entered in line 1480. The ID$ string is formed by selecting the A$(I(1),0) word from the number conversion matrix. The I(1) portion of the argument specifies the row, and the 0 specifies the column. The "1" in I(1) specifies the digit position in the string. After completing the ID$ string, the program continues in line 1780, where the Y$ string is formed by concatenating ID$+IC$ plus the stars on each side of the printed amount. The string is tabbed and printed in line 1790, after which the program reenters the Check Printing loop.

A word of caution. Be sure you clearly understand the portion of the

Table II

No.	Program Name	Core Req.*
10 CRAS-SF	Suspense Files	7949
11 CRAS-CP	Check Printer	7105
12 CRAS-TRF	New Disk Transfer	7982

*Passive state. Does not include random file buffer reservations during initialization.

This table shows the core memory requirements of the CRAS System Programs covered in this article.

program just described dealing with a single-digit dollar amount before you proceed to the subroutine dealing with a double-digit dollar amount (lines 1510 to 1540). Similarly, understand this portion before proceeding to the remaining subroutines. As you'll see, the subroutines become progressively more complex.

I see no way of significantly simplifying number-to-word conversions of this type and welcome your suggestions if you know a better way. As you can see, the number of statements required grows exponentially with the length of the dollar amount. The method of forming the string, ID$, varies greatly with the presence and location of zeros, and whether "teens" are involved.

For hundreds and thousands, the first element of the string is IT$. For example, in 1570, if the second and third digits from the left are zeros (as in the number 100), and since L=3, then the number must be in the series 100, 200 . . . 900. Thus ID$ will equal IT$. IT$, in turn, will equal A$(I(1),0)+"HUNDRED". I(1) in the expression refers to the value of the first digit in the string, the line number of the string conversion matrix, and the zero refers to the column number. Thus, for the number "200", (I(1),0) produces a "TWO" from the string conversion matrix. When concatenated, as in 1560 to 1570, the string becomes "TWO HUNDRED". Careful study should make the balance of the subroutines understandable.

Note that the check number, N4, is incremented in line 940, and this value is translated to a string, A7$, in line 900. But again the conversion produces a preceding null in the string. For example, "1234" becomes " 1234". If the null were not deleted, the final digit of a four-digit check number would be truncated because of field definitions in the transaction entry record. " 1234" would become " 123". The null is discarded by the convenient MID$ (STR$(N4), 2) in line 900.

After the Check Print loop has cycled Q1-P times, the program exits to the CRT and runs CRAS-RS, the Routine Select Program.

CRAS-TRF (Program 12)

Essential requirements of a program to permit the continuation of CRAS onto new disks include the following:

- Transfer initialization files (such constants as name of the checking account, etc.).
- Transfer account ending balances (such as personal accounts, business accounts, check register balances, etc.).
- Transfer the suspense file.
- File the first month number (FM) of the new disk, and the last transaction number (LQ) of the old disk, thus permitting continuation of the existing transaction number sequence, even though record numbers on the new disk will increment from "1".

PUT and GET records will now be referenced as record numbers Q-LQ and J-LQ, respectively. Further, the FIELD, RSET and string-to-digits conversions will have to be altered to include the additional variables on the new disk. These program changes are noted in Table III.

However, be careful to use the revised programs only on new disks. Continue to use the original programs on the old disk, with the obvious exception of the Transaction Entry Program, which should no longer be used on the first disk.

The program gathers the required data from existing files, UNLOADs (line 1810), then requests you to insert the new disk in the drive. The disk is then MOUNTed under program control.

Next, input the number of the new month, FM (line 1930), and the program then files FM together with all the data collected from the old disk (lines 2020 to 2150).

The program concludes with a reminder of the changes you must make to the CRAS System programs for use on the new disk.

CRAS-TRF is not called by CRAS-RS, but must be loaded directly. This precaution prevents inadvertently closing files. Sample Run 3 shows the material presented on the CRT. □

Inquires concerning the CRAS system should be addressed to Personal Computing; or to O.E. Dial, GSPA, University of Colorado, Room 205, Old Armory Building, Boulder, CO 80309.

Sample Run III

```
* * * NEW DISK PROGRAM * * *

THIS PROGRAM FUNCTIONS TO TRANSFER FIXED INITIALIZATION DATA AND CURRENT
BALANCES FROM ONE DISK TO ANOTHER.  THE DISKS WILL BE REFERRED TO AS THE OLD
DISK AND THE NEW DISK LATER ON.

THIS PROGRAM WILL CLOSE OUT THE OLD DISK INSOFAR AS ENTERING NEW TRANSACTIONS
ARE CONCERNED.  THE OLD DISK MAY BE USED ONLY FOR PURPOSES OF PRINTING THE
VARIETY OF FINANCIAL STATEMENTS AVAILABLE.

DO YOU WISH TO CONTINUE? YES

INITIALIZATION DATA, ACCOUNT BALANCES, AND THE SUSPENSE FILE ARE NOW BEING
READ.

OK, ALL NEEDED DATA IS NOW IN CORE MEMORY AND READY TO BE TRANSFERRED TO THE
NEW DISK.

THE OLD DISK HAS NOW BEEN UNLOADED.  YOU MAY REMOVE IT AND PLACE THE NEW DISK IN
THE DRIVE FOR MOUNTING UNDER PROGRAM CONTROL.

 DON'T FORGET THAT THE NEW DISK MUST HAVE BEEN PRE-INITIALIZED.

PRESS 'RETURN' WHEN READY TO CONTINUE?

YOU HAVE PRESSED RETURN, BUT IF YOU HAVE DONE SO WITH THE OLD DISK STILL IN THE
DRIVE, DAMAGE TO YOUR OLD FILES COULD RESULT.  SO,
TO PLAY IT SAFE, YOU WILL BE ASKED TO PRESS 'RETURN' AGAIN IF READY TO CONTINUE.

PRESS 'RETURN' IF THE NEW DISK IS IS THE DRIVE?

FINE.  THE DATA WILL NOW BE WRITTEN ON THE NEW DISK.

WHAT IS THE NUMBER OF THE FIRST MONTH OF THE NEW DISK? 4

* NOW FILING THE NAMES OF THE CHECKING ACCOUNT AND BANK, AND THE YEAR *

* NOW FILING ENDING TRANSACTION AND MONTH NUMBERS FOR OLD DISK *

* NOW FILING THE CURRENT ACCOUNT BALANCES *

* NOW FILING THE SUSPENSE FILE ENTRIES *

OK, ALL DATA HAS BEEN FILED ON THE NEW DISK.

YOU ARE REMINDED THAT CERTAIN PROGRAMS OF THE OLD DISK MUST BE MODIFIED BEFORE
BEING RUN ON THE NEW DISK.  (SEE THE TABLE IN THE ATTACHED ARTICLE.)

NOTE, TOO, THAT THE MODIFIED VERSIONS SHOULD BE RUN ONLY ON THE NEW DISK.

A GOOD PROCEDURE WOULD BE TO 'SAVE' A CRASS PROGRAM ON THE NEW DISK, THEN MODIFY
IT AS PER INSTRUCTIONS, THEN 'SAVE' IT AGAIN ON THE NEW DISK.  THEN EACH REMAIN-
ING PROGRAM SHOULD BE SIMILARLY TREATED.

WHEN THIS IS DONE, THE SYSTEM IS READY TO FUNCTION ON THE NEW DISK.

PLEASE NOTE THAT THIS PROGRAM DOES NOT CHAIN BACK TO PROGRAM 3 'ROUTINE SELECT'.

OK
```

Table III

PROGRAM CHANGES REQUIRED FOR SYSTEM CONTINUATION ON NEWER DISKS
NOTE: THE ACTUAL FILE NUMBER BEING USED SHOULD BE SUBSTITUTED FOR 'X'

PROGRAM	STATEMENT	
1		NO CHANGE
2		NO CHANGE
3		NO CHANGE
4	1390	PUT #X, Q~LQ
4	1480	PUT #X, Q~LQ
4	1560	PUT #X, Q~LQ
4	1830	FIELD #X, 2 AS M1$, 2 AS Q1$, 2 AS QM$, 2 AS QJ$, 2 AS LQ$, 2 AS FM$, 116 AS DU$
4	1850	Q1=CVI(Q1$): M1=CVI(M1$): QM=CVI(QM$): QK=CVI(QJ$): LQ=CVI(LQ$): FM=CVI(FM$)
4	2130	FIELD #X, 2 AS M1$, 2 AS Q1$, 2 AS QM$, 2 AS QJ$, 2 AS LQ$, 2 AS FM$, 116 AS DU$
4	2140	RSET M1$=MKI$(M): RSET Q1$=MKI$(Q): RSET QM$=MKI$(QN): RSET QJ$=MKI$(QJ): RSET LQ$=MKI$(LQ): RSET FM$=MKI$(FM)
5	450	FIELD #X, 2 AS M1$, 2 AS Q1$, 2 AS QM$, 2 AS QJ$, 2 AS LQ$, 2 AS FM$, 116 AS DU$
5	470	Q1=CVI(Q1$): M1=CVI(M1$): QM=CVI(QM$): QK=CVI(QJ$): LQ=CVI(LQ$): FM=CVI(FM$)
5	510	GET #X, J~LQ
5	620	GET #X, J~LQ
5	1760	GOSUB 450: GOSUB 310: IF M=FM THEN 1810
5	2030	IF M=FM THEN P=LQ+1: GOTO 2060
6	410	FIELD #X, 2 AS M1$, 2 AS Q1$, 2 AS QM$, 2 AS QJ$, 2 AS LQ$, 2 AS FM$, 116 AS DU$
6	430	Q1=CVI(Q1$): M1=CVI(M1$): QM=CVI(QM$): QK=CVI(QJ$): LQ=CVI(LQ$): FM=CVI(FM$)
6	620	GET #X, J~LQ
6	1400	IF F=FM THEN P=LQ+1: GOTO 1430
6	1460	GOSUB 410: GOSUB 290: IF M=FM THEN 1510
7	470	FIELD #X, 2 AS M1$, 2 AS Q1$, 2 AS QM$, 2 AS QJ$, 2 AS LQ$, 2 AS FM$, 116 AS DU$
7	490	Q1=CVI(Q1$): M1=CVI(M1$): QM=CVI(QM$): QK=CVI(QJ$): LQ=CVI(LQ$): FM=CVI(FM$)
7	660	GET #X, J~LQ
7	1480	IF M=FM THEN P=LQ+1: GOTO 1510
7	1550	GOSUB 470: GOSUB 320: IF M=FM THEN 1600
8	650	FIELD #X, 2 AS M1$, 2 AS Q1$, 2 AS QM$, 2 AS QJ$, 2 AS LQ$, 2 AS FM$, 116 AS DU$
8	670	Q1=CVI(Q1$): M1=CVI(M1$): QM=CVI(QM$): QK=CVI(QJ$): LQ=CVI(LQ$): FM=CVI(FM$)
8	950	GET #X, J~LQ
8	1550	GET #X, J~LQ
8	1590	GET #X, J~LQ
8	2220	IF M=FM THEN P=LQ+1: GOTO 2250
8	2280	GOSUB 640: GOSUB 300: IF M=FM THEN 2330
9	420	PUT #X, Q~LQ
9	470	FIELD #X, 2 AS M1$, 2 AS Q1$, 2 AS QM$, 2 AS QJ$, 2 AS LQ$, 2 AS FM$, 116 AS DU$
9	490	Q1=CVI(Q1$): M1=CVI(M1$): QM=CVI(QM$): QK=CVI(QJ$): LQ=CVI(LQ$): FM=CVI(FM$)
9	680	GET #X, J~LQ
9	740	FIELD #X, 2 AS M1$, 2 AS Q1$, 2 AS QM$, 2 AS QJ$, 2 AS LQ$, 2 AS FM$, 116 AS DU$
9	750	RSET M1$=MKI$(M): RSET Q1$=MKI$(Q): RSET QM$=MKI$(QN): RSET QJ$=MKI$(QJ): RSET LQ$=MKI$(LQ): RSET FM$=MKI$(FM)
9	1360	Q1=CVI(Q1$): M1=CVI(M1$): QM=CVI(QM$): QK=CVI(QJ$): LQ=CVI(LQ$): FM=CVI(FM$)
10		NO CHANGE
11	380	PUT #X, Q~LQ
11	430	FIELD #X, 2 AS M1$, 2 AS Q1$, 2 AS QM$, 2 AS QJ$, 2 AS LQ$, 2 AS FM$, 116 AS DU$
11	450	Q1=CVI(Q1$): M1=CVI(M1$): QM=CVI(QM$): QK=CVI(QJ$): LQ=CVI(LQ$): FM=CVI(FM$)
11	500	GET #X, J~LQ

```
10 '     ************************************************************************************************************
20 '     *                                                                                                          *
30 '     *                         CRAS (PROGRAM 10): CHECK REGISTER ACCOUNTING SYSTEM - O. E. DIAL  1978          *
40 '     *                                                                                                          *
50 '     ************************************************************************************************************
60 '
70 '
80 '     **********************************************
90 '     *
100 '    *        CRAS-SF: SUSPENSE FILE
110 '    *
120 '    **********************************************
130 '
140 CLEAR 2500: WIDTH 132
150 DEFINT I-T
160 DEFINT Y
170 WE$="##    \

      \"
180 DATA JANUARY, FEBRUARY, MARCH, APRIL, MAY, JUNE, JULY, AUGUST, SEPTEMBER, OCTOBER, NOVEMBER, DECEMBER
190 DIM M$(12), D2(11), S$(11)
200 '
210 FOR I=1 TO 12
220     READ M$(I)'                                 * * READ THE CALENDAR
230 NEXT I
240 '
250 '
260 OPEN "R", 1, "BT-INIT1"'                        * * GET THE NAMES OF THE CHECKING ACCOUNT/BANK
270     FIELD #1, 2 AS IM$, 4 AS RY$, 122 AS NM$
280     GET #1, 1
290     IM=CVI(IM$): YR$=RY$: B$=NM$
300 CLOSE
310 '
320 FOR J1=1 TO 126
330     IF MID$(B$, J1, 1)=" " THEN 350'            * * STRIP AWAY THE NULLS AT THE BEGINNING OF THE STRING
340     GOTO 360
350 NEXT J1
360 B$=MID$(B$, J1)
370 '
380 PRINT CHR$(12)
```

```
390 PRINT TAB(20)"SUSPENSE FILE ROUTINE": PRINT: PRINT
400 '
410 '           * * * ROUTINE SELECTION (MENUE)
420 '
430 Y=0: INPUT "SELECT: PRINT (1); ADD (2); DELETE (3); OR END ROUTINE (4)"; Y: PRINT
440     IF Y<1 OR Y>4 THEN 430
450     ON Y GOTO 530, 1350, 1650, 460
460 CLOSE
470 PRINT CHR$(12)
480 RUN "CRAS-RS"'                              * * RETURN TO 'PROGRAM SELECTION'
490 '
500 '
510 '           * * * TO PRINT THE SUSPENSE FILE
520 '
530 PRINT CHR$(12)
540 PRINT TAB(20)"PRINT SUSPENSE FILE ROUTINE": PRINT: PRINT
550 GOSUB 590 '                                 * * TO PRINT THE STATEMENT HEADING
560 GOSUB 820'                                  * * TO PRINT THE SUSPENSE FILE
570 GOTO 380'                                   * * RETURN TO THE 'MENUE'
580 '
590 GOSUB 1920'                                 * * TO SHIFT PORTS TO THE PRINTER
600     GOSUB 1990'                             * * PRINT DOUBLE UNDERSCORE
610     Y$="SUSPENSE FILE FOR " + YR$'          * * FORM THE NAME OF THE STATEMENT STRING
620     GOSUB 2050'                             * * CENTER AND PRINT THE STRING
630     Y$=B$'                                  * * FORM THE NAME OF ACCOUNT/BANK STRING
640     GOSUB 2050'                             * * CENTER AND PRINT THE STRING
650     GOSUB 2020'                             * * PRINT A SINGLE UNDERSCORE
660 RETURN
670 '
680 '           * * * FILING SUBROUTINES
690 '
700 FIELD #2, 2 AS LD$, 126 AS US$'             * * TO PUT THE SUSPENSE FILE
710     RSET LD$=MKI$(LD): LSET US$=SU$
720     PUT #2, 10*J+K'                         * * 'J' IS THE MONTH, AND 'K' IS THE NUMBER OF THE ITEM WITHIN THE MONTH
730 RETURN
740 '
750 FIELD #2, 2 AS LD$, 126 AS US$'             * * TO GET THE SUSPENSE FILE
760     GET #2, 10*J+K'                         * * 'J' IS THE MONTH, AND 'K' IS THE NUMBER OF THE ITEM WITHIN THE MONTH
770     LD=CVI(LD$): SU$=US$
780 RETURN
790 '
800 '           * * * PROGRAM CONTINUATION
810 '
820 OPEN "R", 2, "BT-SUSP1"'                    * * TO PRINT THE BODY OF THE STATEMENT
830     LN=58: PG=2'                            * * 'LN' IS THE LINE COUNTER, AND 'PG' IS THE PAGE NUMBER
840 '
850     FOR J=0 TO 11
860             PRINT M$(J+1)'                  * * PRINT THE NAME OF THE MONTH
870 '
880             FOR J3=1 TO LEN(M$(J+1))
890                     PRINT "-";'             * * UNDERSCORE THE SUBTITLE (MONTH NAME)
900             NEXT J3: PRINT: PRINT: LN=LN-3' * * UPDATE THE LINE COUNT
910 '
920             FOR K = 1 TO 10'                * * TEN MESSAGES PER MONTH MAXIMUM
930                     GOSUB 750'              * * GET THE MESSAGE
940                     IF LD=0 THEN 960'       * * LAST MESSAGE IN MONTH FLAG (LD=0)
950                     D2(K)=LD: S$(K)=SU$'    * * D2(K) IS THE MESSAGE DATE ARRAY; S$(K) IS THE MESSAGE ASSOCIATED WITH THAT DATE
960             NEXT K
970 '
980             J4=9'                   * * THE CRONOLOGICAL SORT ROUTINE
990             FOR J2=1 TO J4: LF=0: J4=J4-1'          * * 'LF' IS THE SWAP FLAG
1000 '                                  * * 'J4' LOWERS THE NUMBER OF THE TOP RECORD TO BE SORTED WITH EACH PASS
1010 '
1020                    FOR K2=1 TO 9
1030                            IF D2(K2)=0 THEN 1080
1040                            IF D2(K2)<D2(K2+1) THEN 1080
1050                            SWAP D2(K2), D2(K2+1)
1060                            SWAP S$(K2), S$(K2+1)
1070                            LF=1
1080                    NEXT K2: IF LF=0 THEN 1130'     * * TEST THE SWAP FLAG
1090 '
1100            NEXT J2
1110 '
1120 '                                          * * PRINT THE MESSAGES FOR THE MONTH IN CRONOLOGICAL ORDER
1130            FOR J2=1 TO 10
1140                    IF D2(J2)=0 THEN 1170
1150                    PRINT USING WE$; D2(J2), S$(J2):LN=LN-1'        * * PRINT THE DAY AND THE MESSAGE
1160                    D2(J2)=0: S$=""'        * * INITIALIZE THE DATA
1170            NEXT J2: PRINT: PRINT: LN=LN-2
1180 '
1190            IF LN>14 THEN 1250'             * * SAVE 14 LINES FROM THE BOTTOM OF THE PAGE
1200            PRINTCHR$(12)'                  * * ADVANCE TO THE NEXT PAGE
1210            PRINT TAB(95) "SUSPENSE FILE FOR "YR$", PAGE "PG'       * * PRINT A SUMMARY TITLE
1220            PG=PG+1'                        * * INCREMENT THE PAGE COUNTER
1230            LN=60'                          * * INITIALIZE THE LINE COUNTER
1240            GOSUB 2020'                     * * PRINT A SINGLE UNDERSCORE
1250    NEXT J'                                 * * GO BACK FOR THE NEXT MONTH
1260 '
1270    CLOSE
1280    GOSUB 2020'                             * * PRINT A SINGLE UNDERSCORE
1290    PRINT CHR$(12)'                         * * ADVANCE TO TOP OF FORM
1300    CONSOLE16,0'                            * * CHANGE PORTS TO CRT
1310 RETURN
1320 '
1330 '          * * * TO ADD TO THE SUSPENSE FILE
1340 '
1350 PRINT CHR$(12)
1360 PRINT TAB(20) "ADD TO THE SUSPENSE FILE ROUTINE": PRINT: PRINT
1370 OPEN "R", 2, "BT-SUSP1"'                   * *
1380    PRINT "ENTER A ZERO FOR THE NUMBER OF THE MONTH WHEN YOU HAVE FINISHED.": PRINT
1390    INPUT "WHAT MONTH NUMBER"; J: J=J-1: PRINT
1400    IF J>11 THEN PRINT "INPUT ERROR": PRINT: GOTO 1390
1410    IF J<0 THEN 1600'                       * * SIGNALS THE END OF MESSAGE INPUTTING
1420    LF=0'                                   * * 'LF' IS THE 'ALL SPACE CONSUMED' FLAG
1430    GOSUB 1460'                             * * TO FIND A RECORD NOT IN USE
1440    IF LF=1 THEN 1380 ELSE GOTO 1510'       * * NO MORE SPACE AVAILABLE FOR THAT MONTH IF LF=1
1450 '
1460    FOR K = 1 TO 10'                        * * TO FIND THE NUMBER OF AN UNUSED RECORD IN THAT MONTH
1470            GOSUB 750
1480            IF LD=0 THEN RETURN
```

```
1490    NEXT K: PRINT "ALL MESSAGE SPACE FOR "M$(J+1)" HAS BEEN CONSUMED.": PRINT: LF=1: RETURN
1500 '
1510    PRINT "YOU ARE ALLOWED A MAXIMUM OF 10 MESSAGES PER MONTH.": PRINT
1520    PRINT "ENTER A ZERO WHEN YOU HAVE FINISHED FOR THE MONTH.": PRINT
1530    INPUT "NUMBER OF DAY  "; LD: PRINT
1540            IF LD=0 THEN 1380
1550    PRINT "MESSAGE:": PRINT
1560    LINE INPUT SU$: PRINT: PRINT
1570    GOSUB 700'                              * * FILE THE MESSAGE
1580    GOSUB 1460
1590    IF LF<>1 THEN 1530 ELSE 1380
1600    CLOSE
1610 GOTO 380'                                  * * RETURN TO THE 'MENUE'
1620 '
1630 '          * * * TO DELETE FROM THE SUSPENSE FILE
1640 '
1650 PRINTCHR$(12)
1660 PRINT TAB(20)"DELETE FROM THE SUSPENSE FILE ROUTINE": PRINT: PRINT
1670 OPEN "R", 2, "BT-SUSP1"
1680    PRINT "ENTER A 'ZERO' FOR THE NUMBER OF THE MONTH WHEN FINISHED.": PRINT
1690    INPUT "NUMBER OF THE MONTH  "; J: J=J-1: PRINT
1700    IF J>11 THEN PRINT "INPUT ERROR": PRINT: GOTO 1690
1710    IF J<0 THEN 1870'               * * SIGNALS THE END OF MESSAGE DELETIONS
1720    PRINT "EACH ENTRY FOR THE MONTH WILL BE PRESENTED TOGETHER WITH THE "
1730    PRINT "DELETE OPTION.": PRINT
1740 '
1750    FOR K = 1 TO 10'                        * * DELETION LOOP
1760            GOSUB 750'                      * * GET THE MESSAGE
1770            IF LD=0 THEN 1840'              * * TEST TO SEE IF THE RECORD IS USED
1780            PRINT LD TAB(3) SU$'            * * DISPLAY THE MESSAGE
1790            Y$="": INPUT "DELETE"; Y$: PRINT'      * * THE DELETE OPTION
1800            IF LEFT$(Y$,1)<>"Y" THEN 1840
1810            PRINT "THE ITEM HAS BEEN DELETED.": PRINT
1820            LD=0: SU$=""'                   * * INITIALIZE THE RECORD
1830            GOSUB 700'                      * * REFILE THE RECORD
1840    NEXT K
1850 '
1860 GOTO 1680
1870    CLOSE
1880 GOTO 380'                                  * * RETURN TO THE MENU
1890 '
1900 '          * * * VARIOUS SUBROUTINES
1910 '
1920 CONSOLE 18,0'                              * * TO SHIFT PORTS TO THE PRINTER
1930    PRINT CHR$(12)'                         * * TO ADVANCE THE PAPER TO 'TOP OF FORM'
1940    CONSOLE 16, 0'                  * * TO SHIFT PORTS TO THE CRT
1950    INPUT "PAPER POSITIONED "; Y$
1960    CONSOLE 18,0'                           * * BACK TO THE PRINTER
1970 RETURN
1980 '
1990 FOR J2=1 TO 131: PRINT"=";: NEXT J2: PRINT: PRINT' * * PRINT A DOUBLE UNDERSCORE
2000 RETURN
2010 '
2020 FOR J2=1 TO 131: PRINT"-";: NEXT J2: PRINT:'* * PRINT A SINGLE UNDERSCORE
2030 RETURN
2040 '
2050 PRINT TAB(INT((132-LEN(Y$))/2)) Y$: PRINT' * * CENTER AND PRINT THE 'Y$' STRING
2060 RETURN
2070 '
OK
```

```
10 '    ************************************************************************************************************************
20 '    *                                                                                                                      *
30 '    *                        CRAS (PROGRAM 11): CHECK REGISTER BASED ACCOUNTING SYSTEM - O. E. DIAL   1978                 *
40 '    *                                                                                                                      *
50 '    ************************************************************************************************************************
60 '
70 '    ************************************************
80 '    *
90 '    *       CRAS-CP: CHECK PRINTER
100 '   *
110 '   ************************************************
120 '
130 CLEAR 500: WIDTH 132
140 DEFINT I-T
150 CLOSE: Q=0: QN=0
160 WC$="**$,###.##"'                           * * STAR FILLED AMOUNT FORMAT
170 DIM M$(12), A$(9,2)
180 '
190 DATA JANUARY, FEBRUARY, MARCH, APRIL, MAY, JUNE, JULY, AUGUST, SEPTEMBER, OCTOBER, NOVEMBER, DECEMBER
200 '
210 FOR I=1 TO 12'                              * * READ THE CALENDAR
220     READ M$(I)
230 NEXT I
240 '
250 OPEN "R", 1, "BT-INIT1"'                    * * GET THE INITIALIZATION DATA
260     FIELD #1, 2 AS IM$, 4 AS RY$, 122 AS NM$
270     GET #1, 1
280 .   IM=CVI(IM$): YR$=(RY$): B$=NM$
290 CLOSE
300 '
310 GOTO 560
320 '
330             * * * FILING SUBROUTINES
340 '
350 OPEN "R", 2, "BT-TRAN1"'                    * * PUT TRANSACTION DATA (CHECK NUMBER)
360     FIELD #2, 2 AS FM$, 2 AS F1$, 1 AS F2$, 1 AS FX$, 1 AS F4$, 30 AS F5$, 30 AS F6$, 4 AS F7$, 2 AS F9$, 4 AS FA$, 1 AS FB$, 50

 AS DU$
370     RSET FM$=MKI$(M): RSET F1$=A1$: LSET F2$=A2$: LSET FX$=X$: LSET F4$=A4$: LSET F5$=A5$: LSET F6$=A6$:  RSET F7$=A7$: RSET F9$

=MKI$(QN): RSET FA$=MKS$(A1): LSET FB$=K6$
380     PUT #2, Q
390     CLOSE 2
400 RETURN
410 '
```

```
420 '                                           * * PUT LAST TRANSACTION AND LAST MONTH NUMBER
430 FIELD #3, 2 AS M1$, 2 AS Q1$, 2 AS QM$, 2 AS QJ$, 120 AS DU$
440     GET #3, 1
450     Q1=CVI(Q1$): M1=CVI(M1$): QM=CVI(QM$): QJ=CVI(QJ$)
460 RETURN
470 '
480 '                                           * * GET TRANSACTION DATA (CHECK FLAG)
490 FIELD #2, 2 AS FM$, 2 AS F1$, 1 AS F2$, 1 AS FX$, 1 AS F4$, 30 AS F5$, 30 AS F6$, 4 AS F7$, 2 AS F9$, 4 AS FA$, 1 AS FB$, 50 AS

DU$
500     GET #2,  J
510     A1=CVS(FA$): M=CVI(FM$): QN=CVI(F9$): A1$=F1$: A7$=F7$: A2$=F2$: X$=FX$: A4$=F4$: A5$=F5$: A6$=F6$: K6$=FB$
520 RETURN
530 '
540 '           * * * PROGRAM CONTINUATION
550 '
560 PRINT CHR$(12)'                         * * TO CLEAR THE CRT SCREEN
570 PRINT TAB(10) "* * CHECK PRINTER ROUTINE * *": PRINT: PRINT
580     INPUT "WHAT IS THE FIRST TRANSACTION NUMBER FOR WHICH YOU WANT CHECKS PRINTED"; P: PRINT
590             IF P<1 THEN PRINT "INPUT ERROR": PRINT: GOTO 580
600     INPUT "WHAT IS THE NUMBER OF THE FIRST CHECK TO BE PRINTED"; N4: PRINT
610             IF N4<1 THEN PRINT "INPUT ERROR": PRINT: GOTO 600
620 '
630     OPEN "R", 3, "BT-LMLT1"
640             GOSUB 430'                  * * GET LAST MONTH AND LAST TRANSACTION NUMBER
650     CLOSE
660 '
670     IF P>Q1 THEN PRINT "INPUT ERROR.  YOUR LAST TRANSACTION NUMBER IS "Q1".": PRINT: GOTO 580
680 '
690     INPUT "CHECKS POSITIONED";Y$: PRINT
700     CONSOLE 18, 0'                      * * SHIFT PORTS TO PRINTER
710 '
720     GOSUB 1010'                         * * READ THE NUMBER STRINGS
730 '
740 '           * * * CAUTION: THIS PORTION OF THE PROGRAM MUST BE TAILORED TO THE FORMAT OF THE CHECKS YOU WILL USE.
750 '                  PARTICULAR REFERENCE IS MADE TO TABS, INDEXES, AND THE LOCATION OF DATA ON THE CHECK.
760 '
770     FOR J = P TO Q1:Q=J
780             K6$=""
790             OPEN "R", 2, "BT-TRAN1"
800             GOSUB 490: CLOSE'           * * GET TRANSACTION DATA
810             IF K6$<>"*" THEN 950'       * * 'K6$' IS THE 'CHECK ALREADY PRINTED' FLAG
820             IF A7$="*****" THEN 840'    * * 'A7$' IS THE CHECK NUMBER.  IF TO BE PRINTED, WILL EQUAL "*****"
830             CONSOLE 16,0: PRINT "CHECK NUMBER "A7$" HAS ALREADY BEEN WRITTEN ON TRANSACTION "J".": PRINT: CONSOLE 18,0: GOTO 950

840             KJ=4: GOSUB 1820'           * * TO INDEX THE PRINTER 'KJ' TIMES
850             PRINT TAB(48) M$(M)" "A1$ TAB(61)MID$(YR$,3): PRINT: PRINT'     PRINT THE DATE BUT ONLY THE LAST TWO DIGITS OF THE Y

EAR
860             PRINT TAB(11) A5$ TAB(66);: PRINT USING WC$; A1: PRINT' * * PRINT THE PAYEE AND THE AMOUNT (IN DIGITS)
870             GOSUB 1230: PRINT: PRINT'   * * FORM AND PRINT THE 'AMOUNT' STRING (IN WORDS)
880             PRINT TAB(12)A6$'           * * NOTE THAT THE AMOUNT OF THE TAB WILL VARY WITH THE CHECK DESIGN
890             KJ=7: GOSUB 1820'           * * INDEX THE PRINTER 'KJ' TIMES
900             A7$=MID$(STR$(N4),2)'       * * IN CONVERTING A NUMBER TO A STRING THE FIRST (LEFT) SYMBOL WILL BE A NULL.
910 '                                       * * THE USE OF 'MID$' DELETES THE NULL
920             GOSUB 350'                  * * TO PUT THE CHECK NUMBER ON FILE
930             A1$="": A2$="": X$="": A4$="": A5$="": A6$="": A7$="": K6$=""'  * * INITIALIZE ALL VARIABLES
940             N4=N4+1'                    * * INCREMENT THE CHECK NUMBER
950     NEXT J
960 '
970     KJ=8: GOSUB 1820'                   * * TO INDEX THE PRINTER 'KJ' TIMES
980     CONSOLE16,0'                        * * RETURN TO THE CRT
990 RUN "CRAS-RS"
1000 '
1010 FOR J = 0 TO 9
1020     FOR K = 0 TO 2
1030             READ A$(J,K)
1040     NEXT K
1050 NEXT J
1060 RETURN
1070 '
1080 '                                      * * THE NUMBER CONVERSION DATA FILE
1090 DATA ,TEN,
1100 DATA ONE,ELEVEN,
1110 DATA TWO,TWELVE,TWENTY
1120 DATA THREE,THIRTEEN,THIRTY
1130 DATA FOUR,FOURTEEN,FORTY
1140 DATA FIVE,FIFTEEN,FIFTY
1150 DATA SIX,SIXTEEN,SIXTY
1160 DATA SEVEN,SEVENTEEN,SEVENTY
1170 DATA EIGHT,EIGHTEEN,EIGHTY
1180 DATA NINE,NINTEEN,NINETY
1190 '
1200 '                                      * * THE NUMBER CONVERSION SUBROUTINE
1210 '                                      * * SEE TEXT FOR DISCUSSION
1220 '
1230 ID=A1: IC=(A1-ID+5E-03)*100
1240     ID$=STR$(ID)
1250     IF IC=0 THEN IC$=" AND NO/100" ELSE IC$=" AND"+STR$(IC)+"/100"
1260     IF ID=0 THEN ID$=" NONE ": GOTO 1780
1270     ID$=STR$(ID)
1280     L=LEN(ID$)
1290     IF L<6 THEN 1350 ELSE CONSOLE 16,0
1300     PRINT "THE VALUE OF THE CHECK EXCEEDS THE PROGRAM LIMIT.": PRINT
1310     PRINT "ALL ENTRIES ON THE CHECK WILL BE PRINTED EXCEPT THE STRING AMOUNT.": PRINT
1320     CONSOLE 18,0: PRINT : PRINT
1330     GOTO 1800
1340 '
1350     FOR M = 1 TO L
1360             IF MID$(ID$,M,1)<>" " THEN 1390
1370     NEXT M
1380 '
1390     ID$=MID$(ID$,M): L=L-M+1
1400 '
1410     FOR K=1TO L
1420             IK$=MID$(ID$,K,1)
1430             I(K)=VAL(IK$)
1440     NEXT K
1450 '
1460     ON L GOSUB 1480,1510,1560,1640:GOTO 1780
```

```
1470     '
1480             ID$=A$(I(1),0)
1490             RETURN
1500 '
1510             IF ID<20 THEN ID$=A$(I(2),1): GOTO 1540
1520                     IF I(2)=0 THEN ID$=A$(I(1),2)+A$(I(2),0):GOTO 1540
1530                     ID$=A$(I(1),2)+"-"+A$(I(2),0)
1540             RETURN
1550 '
1560             IT$=A$(I(1), 0)+" HUNDRED"
1570                     IF I(2)=0 AND I(3)=0 THEN ID$=IT$: GOTO 1620
1580                     IF I(2)=0 THEN ID$=IT$+" "+A$(I(3),0): GOTO 1620
1590                     IF I(3)=0 AND I(2)=1 THEN ID$=IT$+" "+"TEN": GOTO 1620
1600                     IF I(2)=1 THEN ID$=IT$+" "+A$(I(3),1): GOTO 1620
1610                     ID$=IT$+" "+A$(I(2),2)+"-"+A$(I(3),0)
1620             RETURN
1630 '
1640             IT$=A$(I(1),0)+" THOUSAND"
1650                     IF I(2)=0 AND I(3)=0 AND I(4) = 0 THEN ID$=IT$ : GOTO 1760
1660                     IF I(2)=0 AND I(3)=0 THEN ID$=IT$+" "+A$(I(4),0): GOTO 1760
1670                     IF I(2)=0 AND I(3)=1 THEN ID$=IT$+" "+A$(I(4),1): GOTO 1760
1680                     IF I(2)=0 AND I(4)=0 THEN ID$=IT$+" "+A$(I(3),2): GOTO 1760
1690                     IF I(2)=0 THEN ID$=IT$+" "+A$(I(3),2)+"-"+A$(I(4),0): GOTO 1760
1700                     IF I(3)=0 AND I(4)=0 THEN ID$= IT$+" "+A$(I(2),0)+" HUNDRED ": GOTO 1760
1710                     IF I(3)=0 THEN ID$=IT$+" "+A$(I(2),0)+" HUNDRED " +A$(I(4),0): GOTO 1760
1720                     IF I(4)=0 AND I(3)=1 THEN ID$=IT$+" "+ A$(I(2),0)+" HUNDRED " + "TEN": GOTO 1760
1730                     IF I(4)=0 THEN ID$=IT$+" "+A$(I(2),0)+" HUNDRED "+A$(I(3),2): GOTO 1760
1740                     IF I(3)=1 THEN ID$=IT$+" "+A$(I(2),0)+ " HUNDRED "+A$(I(4),1):  GOTO 1760
1750                     ID$=IT$+" "+A$(I(2),0)+" HUNDRED "+A$(I(3),2)+"-"+A$(I(4),0)
1760             RETURN
1770 '
1780     Y$="* * * * * "+ID$+IC$+" * * * * *"
1790             PRINT TAB(72-LEN(Y$))Y$: PRINT
1800 RETURN
1810 '
1820 FOR JK=1 TO KJ'                                     * * TO INDEX THE PRINTER 'KJ' TIMES
1830     PRINT
1840     NEXT JK
1850 RETURN
```

```
10 '    ***************************************************************************************************
20 '    *                                                                                                 *
30 '    *                    CRAS (PROGRAM 12): CHECK REGISTER ACCOUNTING SYSTEM - O. E. DIAL   1978      *
40 '    *                                                                                                 *
50 '    ***************************************************************************************************
60 '
70 '
80 '    **************************************************
90 '    *
100 '   *       CRAS-TRF: NEW DISK TRANSFER
110 '   *
120 DIM AB(39)
130 '   **************************************************
140 '
150 DIM NA$(6), AD$(6), LD(120), SU$(120)
160 CLEAR 2500
170 DEFINT I-T
180 '
190 DIM AC$(36)
200 DIM G2$(39)
210 DIM AB(39)
220 GOTO 1550
230 '
240 '           * * * FILING SUBROUTINES
250 '
260 OPEN "R", 1, "BT-INIT1"'                    * * PUT NAME OF CHECKING ACCOUNT AND BANK, AND YEAR ON NEW DISK
270     FIELD #1, 2 AS IM$, 4 AS RY$, 122 AS NM$
280     RSET IM$=MKI$(IM): RSET RY$=YR$: RSET NM$=B$
290     PUT #1, 1
300     CLOSE
310 RETURN
320 '
330 OPEN "R", 1, "BT-INIT1"'                    * * GET NAME OF CHECKING ACCOUNT AND BANK, AND YEAR FROM OLD DISK
340     FIELD #1, 2 AS IM$, 4 AS RY$, 122 AS NM$
350     GET #1, 1
360     IM=CVI(IM$): YR$=RY$: B$=NM$
370     CLOSE
380 RETURN
390 '
400 OPEN "R", 2, "BT-INIT2"'                    * * PUT ACCOUNT TITLES ON NEW DISK
410 '
420     FOR J = 1 TO 6 '                        * * SIX TITLES PER RECORD; SIX RECORDS IN THE FILE
430 '
440             FOR K = 1 TO 6
450                     FIELD #2, 2 AS D1$, (K-1)*21 AS D2$, 21 AS NA$(K)
460                     RSET NA$(K)=AC$(K)
470             NEXT K
480 '
490             PUT #2, J
500     NEXT J
510 '
520 RETURN
530 '
540 OPEN "R", 2, "BT-INIT2"'                    * * GET ACCOUNT TITLES FROM OLD DISK
550 '
560     FOR J = 1 TO 6
570 '
580             FOR K = 1 TO 6
590                     FIELD #2, 2 AS D1$, (K-1) * 21 AS D2$, 21 AS AD$(K)
600             NEXT K
610 '
620                     GET #2, J
630 '
640             FOR K = 1 TO 6
650                     AC$(K-6+(J*6))=AD$(K)
660             NEXT K
670 '
```

```
680     NEXT J
690 '
700     CLOSE
710 RETURN
720 '
730 OPEN "R", 3, "BT-LMLT1"'                        * * GET NUMBERS OF LAST MONTH, LAST TRANSACTION, AND
740 '                                                     CONTINOUSLY CLEARED TRANSACTION FROM OLD DISK
750     FIELD #3, 2 AS M1$, 2 AS Q1$, 2 AS QM$, 2 AS QJ$, 120 AS DU$
760     GET #3, 1
770     CLOSE
780     Q1=CVI(Q1$): M1=CVI(M1$): QM=CVI(QM$): QJ=CVI(QJ$)
790 RETURN
800 '
810 OPEN "R", 4, "BT-ACAB1"'                        * * GET BUSINESS ACCOUNTS BALANCES FROM OLD DISK
820     FOR I = 1 TO 16
830             FIELD #4, 64 AS I1$, (I-1) * 4 AS I2$, 4 AS G2$(I)
840     NEXT I
850 '
860     GET #4, M
870 CLOSE
880 '
890 OPEN "R", 5, "BT-ACAB2"'                        * * GET PERSONAL ACCOUNTS BALANCES FROM OLD DISK
900             FOR I = 17 TO 39
910                     FIELD #5, 36 AS I1$, (I-17) * 4 AS I2$, 4 AS G2$(I)
920             NEXT I
930 '
940     GET #5, M
950     CLOSE
960 '
970     FOR I = 1 TO 39'                            * * CONVERT BALANCES FROM STRINGS TO SINGLE PRECISION NUMBERS
980             AB(I)=CVS(G2$(I))
990     NEXT I
1000 '
1010 RETURN
1020 '
1030 OPEN "R", 4, "BT-ACAB1"'                       * * PUT BUSINESS ACCOUNTS BALANCES ON NEW DISK
1040 '
1050    FOR I = 1 TO 16
1060            FIELD #4, 64 AS I1$, (I-1) * 4 AS I2$, 4 AS G2$(I)
1070            RSET G2$(I)= MKS$(AB(I))
1080    NEXT I
1090    PUT #4, M
1100    CLOSE
1110 '
1120 OPEN "R", 5, "BT-ACAB2"'                       * * PUT PERSONAL ACCOUNTS BALANCES ON NEW DISK
1130    FOR I = 17 TO 39
1140            FIELD #5, 36 AS I1$, (I-17) * 4 AS I2$, 4 AS G2$(I)
1150            RSET G2$(I) = MKS$(AB(I))
1160    NEXT I
1170    PUT #5, M
1180    CLOSE
1190 RETURN
1200 '
1210 OPEN "R", 3, "BT-LMLT1"'                       * * PUT ON NEW DISK THE NUMBERS OF:
1220 '                                                    LAST MONTH; LAST TRANSACTION; NOTE;
1230 '                                                    CONTINUOUSLY CLEARED TRANSACTION
1240 '                                               * * SEE TEXT FOR ADDITIONAL DISCUSSION * *
1250    FIELD #3, 2 AS M1$, 2 AS Q1$, 2 AS QM$,  2 AS QJ$, 2 AS LQ$, 2 AS FM$, 116 AS DU$
1260    RSET M1$=MKI$(M): RSET Q1$=MKI$(Q): RSET QM$=MKI$(QN): RSET QJ$=MKI$(QJ): RSET LQ$=MKI$(LQ): RSET FM$=MKI$(FM)
1270    PUT #3, 1
1280    CLOSE
1290 RETURN
1300 '
1310 OPEN "R", 6, "BT-SUSP1"'                       * * PUT THE SUSPENSE FILE ON THE NEW DISK
1320 '
1330    FOR J = 1 TO 120
1340            FIELD #6, 2 AS LD$, 126 AS US$
1350            RSET LD$=MKI$(LD): LSET US$=SU$
1360            PUT #6, 10*J+K
1370    NEXT J
1380 '
1390    CLOSE
1400 RETURN
1410 '
1420 OPEN "R", 6, "BT-SUSP1"'                       * * GET THE SUSPENSE FILE FROM THE OLD DISK
1430 '
1440    FOR J = 1 TO 120
1450            FIELD #6, 2 AS LD$, 126 AS US$
1460            GET #6, J
1470            LD=CVI(LD$): SU$=US$
1480    NEXT J
1490 '
1500    CLOSE
1510 RETURN
1520 '
1530 '          * * * PROGRAM CONTINUTATION * * *
1540 '
1550 PRINT CHR$(12)'                                * * TO CLEAR THE CRT SCREEN
1560 PRINT TAB(20) "* * * NEW DISK PROGRAM * * *": PRINT: PRINT
1570 '
1580 PRINT "THIS PROGRAM FUNCTIONS TO TRANSFER FIXED INITIALIZATION DATA AND CURRENT"
1590 PRINT "BALANCES FROM ONE DISK TO ANOTHER.  THE DISKS WILL BE REFERRED TO AS THE OLD"
1600 PRINT "DISK AND THE NEW DISK LATER ON.": PRINT
1610 PRINT"THIS PROGRAM WILL CLOSE OUT THE OLD DISK INSOFAR AS ENTERING NEW TRANSACTIONS"
1620 PRINT "ARE CONCERNED.  THE OLD DISK MAY BE USED ONLY FOR PURPOSES OF PRINTING THE"
1630 PRINT "VARIETY OF FINANCIAL STATEMENTS AVAILABLE.": PRINT
1640 INPUT "DO YOU WISH TO CONTINUE"; Y$: PRINT
1650    IF LEFT$(Y$,1)="Y" THEN 1670
1660    GOTO 2280
1670 PRINT "INITIALIZATION DATA, ACCOUNT BALANCES, AND THE SUSPENSE FILE ARE NOW BEING"
1680 PRINT "READ.": PRINT: PRINT
1690 GOSUB 730'                                     * * TO GET THE NUMBER OF THE LAST MONTH, LAST TRANSACTION NUMBER, LAST
1700 M=M1
1710 '                                                  NOTE NUMBER, AND LAST CONTINUOUSLY CLEARED TRANSACTION NUMBER.
1720 GOSUB 330'                                     * * TO GET THE NAME OF THE CHECKING ACCOUNT, BANK, AND YEAR
1730 GOSUB 540'                                     * * TO GET THE ACCOUNT TITLES
1740 M=M+1
1750 GOSUB 810'                                     * * TO GET ACCOUNT BALANCES
1760 M=M-1
1770 GOSUB 1420'                                    * * TO GET THE SUSPENSE FILE
```

```
1780 '
1790 PRINT "OK, ALL NEEDED DATA IS NOW IN CORE MEMORY AND READY TO BE TRANSFERRED TO THE"
1800 PRINT "NEW DISK.": PRINT
1810 UNLOAD 0
1820 PRINT "THE OLD DISK HAS NOW BEEN UNLOADED.  YOU MAY REMOVE IT AND PLACE THE NEW DISK IN"
1830 PRINT "THE DRIVE FOR MOUNTING UNDER PROGRAM CONTROL.": PRINT
1840 PRINT " DON'T FORGET THAT THE NEW DISK MUST HAVE BEEN PRE-INITIALIZED.": PRINT
1850 INPUT "PRESS 'RETURN' WHEN READY TO CONTINUE"; Y$: PRINT
1860 PRINT "YOU HAVE PRESSED RETURN, BUT IF YOU HAVE DONE SO WITH THE OLD DISK STILL IN THE"
1870 PRINT "DRIVE, DAMAGE TO YOUR OLD FILES COULD RESULT.  SO,"
1880 PRINT "TO PLAY IT SAFE, YOU WILL BE ASKED TO PRESS 'RETURN' AGAIN IF READY TO CONTINUE.": PRINT"
1890 INPUT "PRESS 'RETURN' IF THE NEW DISK IS IS THE DRIVE"; Y$: PRINT
1900 PRINT "FINE.  THE DATA WILL NOW BE WRITTEN ON THE NEW DISK.": PRINT
1910 '
1920 MOUNT 0
1930 INPUT "WHAT IS THE NUMBER OF THE FIRST MONTH OF THE NEW DISK"; FM: PRINT
1940 Q=Q1: QN=0: LQ=Q1
1950 '
1960 '                                        * * THE NEW VARIABLES ARE:
1970 '                                                LQ=LAST TRANSACTION NUMBER ON OLD DISK
1980 '                                                LN=LAST NOTE NUMBER ON OLD DISK
1990 '                                                LJ=LAST CONTINUOUSLY CLEARED TRANSACTION NUMBER ON OLD DISK
2000 '                                                FM=FIRST MONTH NUMBER OF TRANSACTIONS ON NEW DISK
2010 '
2020 PRINT "* NOW FILING THE NAMES OF THE CHECKING ACCOUNT AND BANK, AND THE YEAR *": PRINT
2030 '
2040 GOSUB 260'                               * * FILE THE NAMES OF THE CHECKING ACCOUNT AND BANK, AND THE YEAR
2050 '
2060 PRINT "* NOW FILING ENDING TRANSACTION AND MONTH NUMBERS FOR OLD DISK *": PRINT
2070 GOSUB 1210'                              * * FILE ENDING TRANSACTION AND MONTH NUMBERS FOR OLD DISK
2080 '
2090 PRINT "* NOW FILING THE CURRENT ACCOUNT BALANCES *": PRINT
2100 M=FM
2110 GOSUB 1030: M=M+1'                               * * FILE CURRENT ACCOUNT BALANCES
2120 GOSUB 1030: M=M-1
2130 '
2140 PRINT "* NOW FILING THE SUSPENSE FILE ENTRIES *":PRINT
2150 GOSUB 1420'                              * * FILE THE SUSPENSE FILE ENTRIES
2160 '
2170 PRINT "OK, ALL DATA HAS BEEN FILED ON THE NEW DISK.": PRINT
2180 PRINT "YOU ARE REMINDED THAT CERTAIN PROGRAMS OF THE OLD DISK MUST BE MODIFIED BEFORE"
2190 PRINT "BEING RUN ON THE NEW DISK.  (SEE THE TABLE IN THE ATTACHED ARTICLE.)": PRINT
2200 PRINT "NOTE, TOO, THAT THE MODIFIED VERSION IS SHOULD BE RUN ONLY ON THE NEW DISK.": PRINT
2210 PRINT "A GOOD PROCEDURE WOULD BE TO 'SAVE' A CRASS PROGRAM ON THE NEW DISK, THEN MODIFY"
2220 PRINT "IT AS PER INSTRUCTIONS, THEN 'SAVE' IT AGAIN ON THE NEW DISK.  THEN EACH REMAIN-"
2230 PRINT "ING PROGRAM SHOULD BE SIMILARLY TREATED.": PRINT
2240 PRINT "WHEN THIS IS DONE, THE SYSTEM IS READY TO FUNCTION ON THE NEW DISK.": PRINT
2250 PRINT "PLEASE NOTE THAT THIS PROGRAM DOES NOT CHAIN BACK TO PROGRAM 3 'ROUTINE SELECT'.": PRINT
2260 '
2270 '
2280 CLOSE
2290 END
```

PERIPHERALS

High Resolution Color Monitors

A series of high-resolution **color video monitors** has been introduced by Amtron Corporation. Designated the 7800 series, the new monitors have been tailored to meet the requirements of broadcasters and those involved in video production and other related fields.

Two models are available: Model 7813, with a high-resolution 13" shadow-mask CRT, and Model 7819, with a 19" CRT packaged in a compact vertical rack space of 15-3/4".

Both models share the same performance specifications as master video monitors, and serve as critical signal evaluation and measurement tools as well. The wideband video processing circuitry appears transparent so that the entire video signal can reach the CRT without abberation.

For more information, contact Amtron Corp., Box number 1160, Aptos, CA 95003; (408) 688-4445. *Circle No. 125.*

RS-232 Serial Interface for AJ 841 I/O ASCII Terminal

Anderson Jacobson, Inc., has a new option for the AJ 841 I/O – an **RS-232 serial interface** with selectable baud rates of 110 to 1200, an 896 character

receive/print buffer and X-on/X-off control characters to and from the computer.

The AJ 841 I/O is also available with a parallel interface that can connect to any S-100 bus microcomputer. Based on the IBM 745 heavy-duty terminal mechanism and electronics by AJ, the AJ 841 I/O can be used on line to a computer or off-line as a typewriter. Both parallel and serial units are available with a choice of EBCD or Correspondence keyboards.

The interface costs $995. For more information, contact Dave Strand at Anderson Jacobson, 521 Charcot Avenue, San Jose, CA 95131; (408) 263-8520. *Circle No. 126.*

Welcome to the new What's Coming Up. We've given our products section a facelift to make it easier to read, more attractive and more useful.

First, we've sorted the new personal computing products into several categories, making it easier for you to locate the particular items you're shopping for. Second, we've added headlines to each product description.

Our products section facelift is just one more step in our continual effort to make Personal Computing *easy to read, fun to peruse, enjoyable to look at and, most important, helpful to you.*

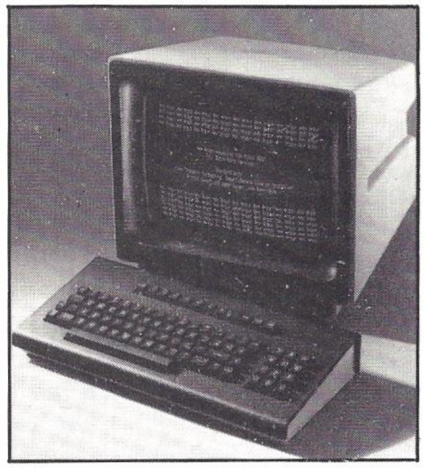
Ann Arbor 531 Terminal Provides Form-Filling Capabilities

The Model 531 **terminal** is intended for business applications requiring operator completion of preprogrammed forms and subsequent data transmission from the terminal to the computer or an auxiliary RS232 printer.

Standard with a memory size of 1920 upper-case ASCII characters displayed in a 24 line by 80 character format, it includes three character accents, formatted data entry, buffered RS232 printer output and edit functions (both character and line). The 531 also has many "extras" such as an 82-key detachable keyboard including a TTY-pad, numeric pad, and function key pad which contains 12 keys used for editing and related terminal functions. It has a 15" nonglare screen and auxiliary video output for daisy-chaining up to ten remote monitors. Baud rates, input/output modes, keyboard and display modes may be selected through the keyboard.

Terminal options include upper/lower case, 40-character line, bell and export power (50 Hz, 220 VAC). A current-loop cable, if desired, converts the RS232 data interface to 20 or 60 MA. A port extender cable allows for interfacing to an auxiliary RS232 printer terminal.

Case options include the standard E-Case, which is the size of a 15-inch



desktop monitor, and can also be mounted in a 19-inch rack through the use of an optional rack panel. The 531 can also be packaged inside Ann Arbor's Design III case, a larger, more streamlined design. Or, it can be purchased without a case for installation into a console or customized packaging by an OEM.

A keyboard option allows the user to specify up to 24 additional function keys for specialized applications.

Unit price for the Model 531E terminal is $1500, with OEM and quantity discounts available. For further information, contact Sarah J. Freeman, Marketing Coordinator, Ann Arbor Terminals, Inc., 6107 Jackson Rd., Ann Arbor, MI 48103; (313) 769-0926. *Circle No. 127.*

New Microprocessor Keyboard Has Dry Reed Switches

A **full-function keyboard** from C.P. Clare uses a single-chip, second generation 8-bit microprocessor that incorporates, in addition to ROM and RAM,

an erasable PROM for software control of all key functions. The keyboard is quickly programmable to exact application requirements. A custom-made keyboard is produced on a rapid design-to-prototype turnaround schedule. The microprocessor also permits automatic repeats, multiple programming of a single board, program changes in the field, serial and/or parallel I/O and N-key rollover.

The dry reed capsule keyswitches are of low-profile design, measuring 1-5/16 inches from base to keytop, and sealed against dust, humidity and hostile environments. They require low operating power and have no power drain. The keytops are non-glare, 2- and 3-shot molded and come in a variety of keytop languages, legends and styles.

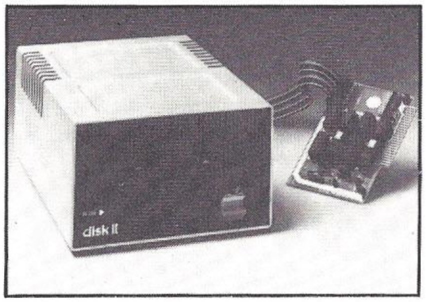
For more information, contact C.P. Clare & Company, 3101 W. Pratt Avenue, Chicago, IL 60645; (312) 262-7700. *Circle No. 133.*

Apple Computer's Disk II

Apple Computer, Inc., announced **Disk II**, an intelligent peripheral for its popular Apple II personal computer. Disk II's home applications include address files, social appointment calendars and recipe files. Personal finance, too, can be implemented with the disk; for example, a user can store a year's worth of financial records in one place and sort them quickly. Likewise, a week's worth of stock prices on the New York Stock Exchange can be stored and processed on a single diskette.

Disk II allows the Apple II to handle business applications including inventory, general ledger and payroll.

The unit's Disk Operating System (DOS) software provides dynamic disk



space allocation and compatibility with existing languages through standard BASIC commands.

The Disk II subsystem consists of an intelligent interface card and either one or two mini-floppy drives. The computer will handle up to seven controller cards and fourteen drives for instant access to more than 1.6 million bytes of data. The combination of a bootstrap loader in ROM and operating system in RAM provides disk handling capability with the following features: full disk capability for systems with as little as 16K bytes of RAM; ability to load and store files by name; random and sequential access; automatically generated file-name directories; storage capacity of 116 kilobytes per diskette; ability to be driven from Apple II power supply with no other power required.

Price, including controller card and drive, is $495. For more information, contact Apple Computer, Inc., 10260 Bandley Dr., Cupertino, CA 95014; (408) 996-1010. *Circle No. 128.*

Bowmar Introduces Flat-Surface Custom Keyboards

Custom designed keyboards from Bowmar Instrument Corporation are available for use with multicolored overlays.

Among the possible applications are data entry devices, test instruments, appliances like microwave ovens and televisions and communications equipment.

In addition to UL recognized parts, Bowmar keyboards feature tactile response for quick and accurate contact, and flat, spill-resistant surfaces for easy maintenance. Designed for any configuration from a basic keyboard to total systems including LEDs or other electronic components, the Keyboards also feature sealed mounting for reliability. Graphic possibilities include legends, colors, sizes, shapes and nomenclatures to meet functional and visual demands.

For more information, contact Bowmar Instrument Corporation, Commercial Products Division, 8000 Bluffton Road, Fort Wayne, IN 46809; (219) 493-4472. *Circle No. 130.*

Charles River Data Offers MF 11/2

Charles River Data now offers its MF-11 LSI-11/**Floppy Disk System** with the DEC LSI-11/2 (KD11-HA) and associated DEC plug-in memory. The MF 11/2 is functionally identical to the PDP 11VO3, but uses only 10-1/2 inches of panel height. The enclosure holds the DEC processor, two Shugart Floppy Disk Drives with controller,

power supply, slides for rack mounting, and the DEC H9270 backpanel. An 8 quad slot backplane is available as an option.

The controller/interface card provides software and media compatibility between the DEC processor and the floppy disk system, which allows use with any of the PDP 11VO3 software packages. It also provides bootstrap loader, self-test and IBM 3740 formatter. The front panel console has the switches for ENABLE/HALT, LINE TIME CLOCK and INITIALIZE. There are also indication lights for "DC POWER" and "RUN". The MF-11/2 is intended for low-end PDP-11VO3 type applications.

For more information, contact Marketing Department, Charles River Data Systems, 4 Tech Circle, Natick, MA 01760; *Circle No. 132.*

PRODUCT INDEX

Centronics Reduces Price for Microprinter

Centronics Data Computer Corp. announced a 33 percent price reduction of their Model P1 **non-impact microprinter** from $595 to $395.

The Model S1 with serial interface was also reduced in price from $695 to $549. Both models are available from Centronics, Hamilton/Avnet and other Centronics distributors.

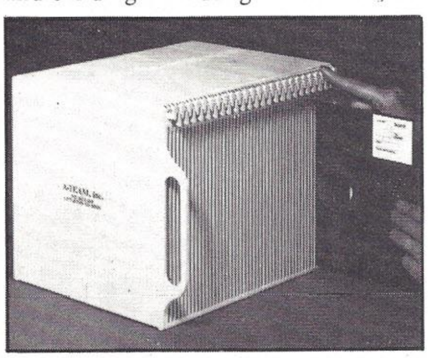
The nonimpact microprinter produces clear dot matrix characters at 240 per second on aluminum coated paper by discharging an electric arc to penetrate the coating. Toners and ribbons are not required. The printed characters are impervious to light, temperature and humidity. In addition, the finished printed page may be reproduced on office copying machines.

The 150 line-per-minute microprinter comes as a complete unit including case, power supply, 96 character ASCII generator and interface, paper roll holder, low paper detector, bell and multi-line asynchronous input buffer. The 150 line per minute unit uses 4-3/4 inch roll paper and provides user software selection of 20, 40 or 80 columns.

Contact Centronics, Hudson, NH 03051; (603) 883-0111. *Circle No. 131.*

A-Team's Floppy Disk Storage System

A-Team, Inc., announced a new floppy disk storage and retrieval system. Their **Floppy Disk Storage System** protects diskettes from coffee spills, cigarette burns, creases, folds and bending. The design offers fully

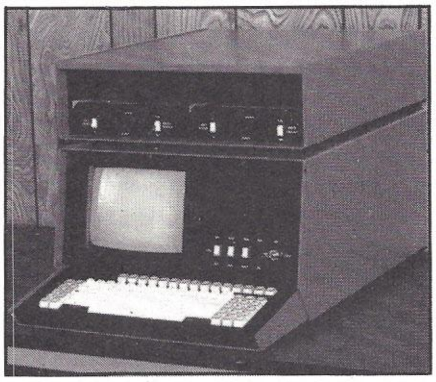


indexed retrieval of diskettes through push button control. The system is lightweight, portable and easily stacked. It can also rest in a file drawer. Systems are available to accommodate 15, 30 and 50 diskettes.

A 15-slot unit costs $69.95; 30-slot, $119.95; 50-slot, $179.95. For more information, contact A-Team, Inc., P.O. Box 719, Broomfield, CO 80020. *Circle No. 129.*

Computer Marketing Announces Multibus/iSBC Compatible Terminal

Computer Marketing, Inc., offers the MB 80, a Multibus/iSBC compatible **intelligent terminal.** Features include a 30 Amp 5 V power supply, 9" CRT, full ASCII keyboard, 2 programmable keypads, 15 function keys, 16 line x 64 character display control-



ler, and choice of processor board from iSBC 80/05 to iSBC 86/12 with addressable memory up to 1 Mbyte.

Seven Multibus slots are provided for system expansion. Available peripherals include semiconductor/core main memory, 5 and 10 Mbyte disk subsystems, floppy disks, mag tape, A/D converters and optically isolated I/O.

Prices start at $2510. Delivery is 60 to 90 days ARO. Contact C.E. Guenther, Product Manager, Computer Marketing, Inc., 257 Crescent Street, Waltham, MA 02154; (617) 894-7000. *Circle No. 135.*

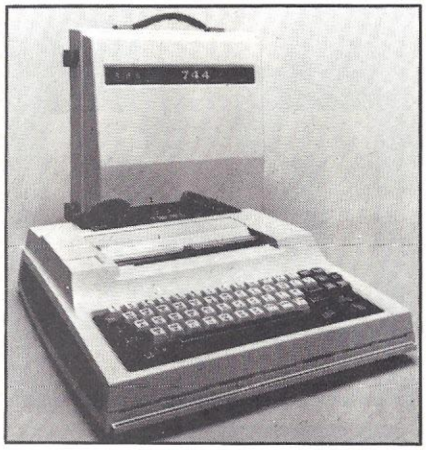
Data Access Systems' DASI 744 Terminal

Data Access Systems, Inc., markets a **portable data terminal**, the DASI 744, a modified TI 743 enhanced with switch selectable parity, EIA RS 232 interface and cable, answer mode and acoustic coupler.

Additional features offer user flexibility and versatility, the company said. The switch selectable parity enables users to communicate with any number of time-sharing services, while

the EIA RS 232 interface and cable allows direct hook-up to minicomputers as well as remote access through the acoustic coupler.

The DASI 744 combines these standard features with microprocessor reliability and solid state printing at operator selectable speeds of 10 and 30



cps. Undesirable noises associated with impact printing are eliminated with the solid state printhead which develops characters on thermal sensitive paper.

The terminal sells for $1595. For more information, contact Data Access Systems, 100 Route 46, Mountain Lakes, NJ 07046. *Circle No. 138.*

CDI Miniterm ASR Terminals Expanded to 32K RAM

Computer Devices Inc. released a **32K Random Access Memory** option to its line of Miniterm 1204 desk-top and Miniterm 1205 Portable ASR terminals. These terminals feature both an integral 32K RAM and a built-in minicassette tape transport with 68,000 characters of removable storage. The increased memory capacity combines with a three-mode keyboard, a 35-character-per-second thermal printer, fully buffered, and switch selectable ASCII or binary code.

The dual storage media of memory and tape work in tandem to provide simultaneous transmit/receive capability, tape-to-memory, memory-to-tape and tape-to-memory-to-line transmission.

In one package weighing less than twenty pounds, the Miniterm ASR units include the following capabilities: 1200 baud transmission rates from tape-, memory-, or keyboard-to-line, allowing off-line data preparation for subsequent transmission at high speeds (up to 1200); versatile three-mode keyboard, operator selected, includes upper/lower case typewriter, TTY, APL, or integral 16-key numeric cluster; 35 characters per second printing speed, fully buffered; compatible with all minicomputers and central processors via EIA Standard RS-232 interface; ASCII or binary code; diagnostic capability for printing control characters; and horizontal and vertical tabbing for easy formatting. High-speed editing capability within the memory provides extensive search, delete, change, append and file control functions.

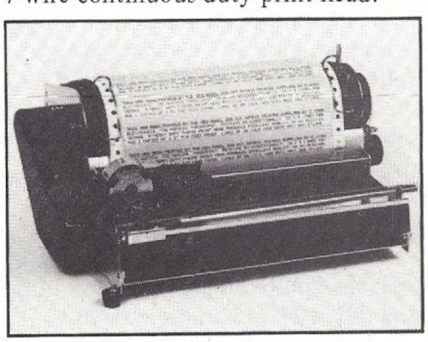
Applications include programming, data-entry/inquiry, data base management, point-to-point message communications and other remote batch data processing tasks.

The 32K option is priced at $750. The base price of the Miniterm 1204 ASR is $3385 including a one year warranty; and $165 per month on a one year lease. For more information, contact Computer Devices, P.O. Box 421, Burlington, MA 01803; (617) 273-1550. *Circle No. 134.*

C. Itoh Develops New 84 lpm Print Mechanism

C. Itoh Electronics, Inc.'s **dot-matrix impact print mechanism** offers 80-column bi-directional printing at low cost. The new OEM printer, with a head life of 100 million characters, was developed for computer terminals, computer output, data loggers and general business applications.

The printing mechanism features a 7-wire continuous-duty print head.

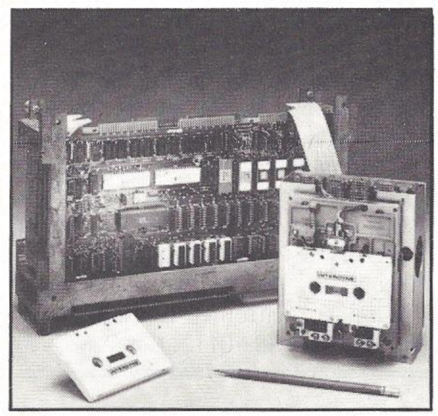


Printing at 84 lines per minute with 80 columns per line, the mechanism produces quality printing on all copies of multi-ply forms. A sprocket paper feed mechanism accepts standard 9.5" wide, pin-feed paper. Loading can be from the bottom or from the rear.

For more information, contact C. Itoh Electronics, Inc., 5301 Beethovan Street, Los Angeles, CA 90066; (213) 390-7778, or 280 Park Avenue, New York, NY 10017; (212) 682-0420; *Circle No. 150.*

SBC Compatible Cassette Interface Model IB 4100

A **digital interface board** compatible with Intel's SBC 80 computer line is offered by Interdyne Company. The board, Model IB 4100, plugs directly into the SBC chassis and has I/O connectors for two Interdyne IC 2500 series cassette drives. Driver software provided in ROM plugs directly into the SBC 80 board. The driver software does the data block formatting and error detection, and reads and writes



to and from memory at 12 IPS (9600 bps), searches at 40 IPS and rewinds at 120 IPS. Other more sophisticated software packages are also available.

Only this card and a cassette drive are needed to add storage capacity of 0.5 megabytes per cassette, a data transfer rate of 1K bytes/sec. and fast memory access.

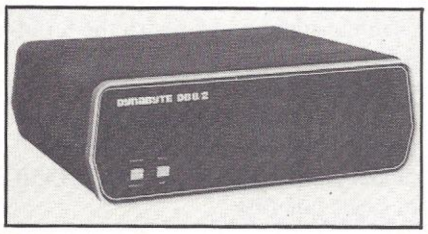
The IB 4100 PC board sells in single quantities for $250, which includes the separate driver ROM. For more information, contact Bill K. Geist, Interdyne, 14761 Califa Street, Van Nuys, CA 91411; (213) 787-6800. *Circle No. 149.*

Dynabyte Computers Implement New Disk Drive Controller Technology

A new line of **microcomputer systems** from Dynabyte features a disk drive controller that increases the choice of disk storage configurations.

Top of the line is the DB8/2 Computer System, which offers up to 1.2

megabytes of mass storage on two 5-inch drives. It uses 77-track Micropolis disk drives and with Dynabyte's new controller offers double or quad density in single or double sided configur-



ations – up to eight times the capacity of single-sided, single-density 5-inch drives.

To implement the drives, Dynabyte developed its Dual Density Floppy Disk Controller, capable of handling a variety of 5-inch and 8-inch drives in dual density on either one or two sides. To permit expansion of the system, the controller can handle up to 16 drives.

The product line was developed for business, professional and scientific applications.

The company's Dynamic Data Compensation yields a double density error rate comparable to single density rates, Dynabyte said.

The DB8/2 includes a 4 MHz Z-80 microprocessor module that also contains two RS232 serial I/O ports, one parallel I/O port, an EPROM programmer, two TMS2716 sockets, vectored interrupts and a real-time clock.

The unit has 32K of RAM and the Disk Controller in a 12-slot backplane fully populated with mil-spec connectors. It uses a regulated power supply designed to comply with U.L. approved standards.

The DB8/2 enclosure incorporates aluminum castings and deckled finish.

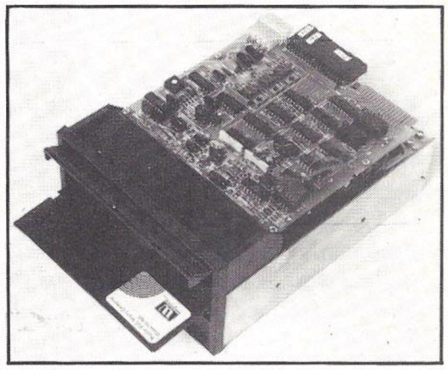
CP/M Disk Operating System is used in the Dynabyte systems. Initial language and software packages from Dynabyte include BASIC, FORTRAN, COBOL, word processing, general ledger and accounts receivable.

Dynabyte also introduced the DB8/1, a Z-80 computer with no mass storage, and the DB8/4 Floppy Disk System with two 8-inch disk drives with up to two megabytes of storage.

A product brochure is available from Dynabyte, Inc., 1005 Elwell Court, Palo Alto, CA 94303; (415) 965-1010. *Circle No. 142.*

Wangco's New Dual-Head, Dual Density Micro Floppy

A new **two-headed micro-floppy disk drive** was recently introduced by the



Wangco Division of The Perkin-Elmer Corporation. The new dual-headed

micro-floppy drive, Model 282, is a compact, low-cost unit designed for mass storage in small systems such as microcomputers, word processors, personal computers and intelligent terminals. According to Wangco officials, the 282 micro-floppy capabilities approach those of standard size disk drives.

In single density operation, using frequency modulation encoding, the 282 micro-floppy stores 1.75 million bits (109.4 kilobytes) of unformatted data per disk side, at a rate of 125,000 bits per second. Using both sides of the diskette in double density MFM encoding, the 282 stores up to 3.5 megabits (437.5 kilobytes) of unformatted data at a rate of 250,000 bps.

Up to four 282 micro-floppy drives can be daisy chained to provide a total on-line storage capacity of 14 megabits. Seek times, track to track, are 25 milliseconds in both the single and double density operating modes.

The 282 micro-floppy drive accepts industry standard diskettes and is compatible with the Shugart SA-450. Gentle handling of the media to protect the data file is assured by both a wide loading door and a positive operator-door interlock feature.

Customer-installed jumper options enable the drive to be used in several different modes, including select drive without head loading; radial head load (load head without selecting drive); drive select for fourth drive; stepper power control option; spindle drive motor controlled by select; and head load with motor on or select.

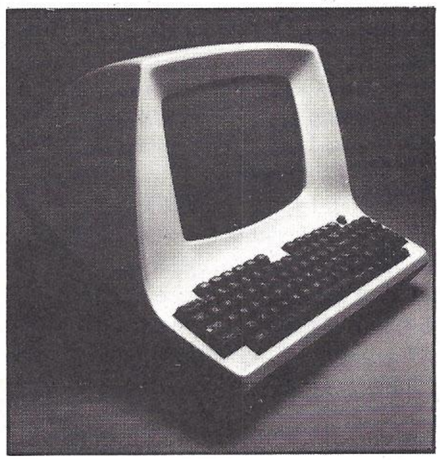
The 282 micro-floppy provides a host system interface via a 0.100" center printed wiring board edge connector, with an industry standard ANSI X3T9.2-compatible pinout for both DC power and signal/control. For more information, contact Wangco Division, 5404 Jandy Place, Los Angeles, CA 90066; (213) 390-8081. *Circle No. 156.*

Lear Siegler Features New Smart CRT Terminal

Lear Siegler, Inc./Data Products Div.'s low-cost **video display terminal** features two pages of memory, function keys and complete editing capabilities. The smart terminal, called the ADM-31, offers users two full 1920 character pages of display with independent page characteristics of protect, write/protect, program mode and cursor retention. If the operator changes to another page, the attributes are automatically stored in memory and are recalled exactly as they were when the page is readdressed.

Completely self-contained, the microprocessor-based ADM-31 comes equipped with keyboard, control logic, character generator, refresh memory and interface.

The terminal's keyboard is integrated with main logic and can generate



a full 128 ASCII character set. It features an integral numeric keypad in a calculator format.

The ADM-31 also includes a function key to transmit a special function sequence to the host. The sequence is SOH, followed by any ASCII keyboard character. All control functions possible from the keyboard can also be executed from the remote computer.

ADM-31's editing capabilities allow the user to clear the screen or use a destructive cursor for character change. On the ADM-31, full or half duplex conversation modes are switch and keyboard selectable. Send line unprotected, send line all, send page unprotected, send page all and send message block transmission modes can also be initiated by the operator or computer. Polling and addressing transmission is optionally available.

For added flexibility, the ADM-31's characteristics are programmed in at the manufacturing stage for compatibility with many industry standard computer systems. The **ADM-31** features a high resolution, 12" diagonal display screen with 1920 easy-to-read characters (24 lines of 80 characters) in a 7 x 9 dot matrix.

The terminal is designed for RS-232C point-to-point and 20 mA current loop interfaces at rates of 50, 75, 110, 134.5, 150, 300, 600, 1200, 1800, 2000, 2400, 3600, 4800, 7200 and 9600 baud. An optional RS-232 extension extends the standard port for chaining additional units to the terminal. A serial printer interface is also optionally available.

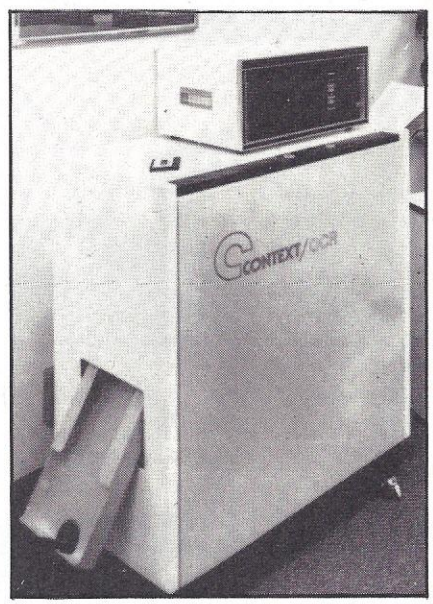
The ADM-31 is also upward compatible with the company's earlier model ADM-1A. It is priced at $1450 in single quantity.

For more information, contact Lear Siegler, Inc./Data Products Division, 714 N. Brookhurst, Anaheim, CA; toll free (800) 854-3805, in CA (714) 774-1010. *Circle No. 151.*

Context Page Reader Converts Typed Pages to WP Media

Context Corporation announced the newest addition to its line of **optical character readers** for word processing. Designated Model 1210-R, the unit can accept typewritten draft copy and convert it to Redactron-compatible cassettes for further editing on a Redactron word processor.

The Context 1210-R, by permitting original typing on regular typewriters rather than on word-processing keyboards, frees the word processor for correcting, editing and final drafting. One word processor can support



the typing production of up to 25 typists. The unit also includes a standard communications interface (RS-232C ASCII), which makes it compatible

with other types of word processing equipment.

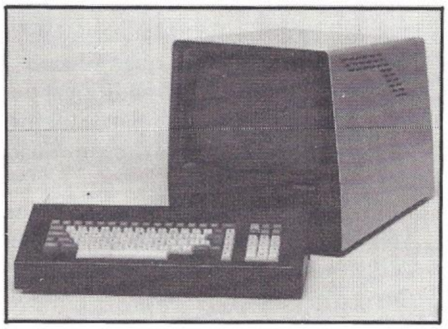
Throughput of the unit is 4 to 5 pages per minute, or over 2000 pages per day. A single 1210-R could support up to 100 draft typists whose work is being edited on up to 10 word processors, the company said. Model 1210-R can read 10 or 12 pitch copy typed in OCRB typefont. Its published rate is below one error in 30,000 characters, or around one error in ten pages.

Price is $19,500. For more information, contact Context Corp., 9 Ray Ave., Burlington, MA 01803; (617) 273-2222. *Circle No. 136.*

New Terminal Features 60 Second Part Replacement

The Teleray Division of Research Inc. recently announced the Series 10 **CRT terminal line**, designed to reduce downtime and service. Any of the terminal's modules (logic, monitor, power and keyboard assemblies) can be replaced as simply as a typewriter ribbon cartridge. Fool-proof mechanical design automatically locks all modules in place with the unit's cover.

The 1061 – first of the Series 10 models – offers programmable features, including function keys, I/O and peripheral speeds, peripheral on/off and wide/narrow character display. It's a



scroll or page mode editing machine with cursor addressing, three-segment clear, insert/delete functions. It transmits either in character mode or by line, message or page. The 1061 operates in format and protect modes with four-level highlighting (dim, blink, inverse video and underlining – programmable in any combination), and in transparent mode (control codes displayed).

Availability of the 1061 is five weeks ARO. It lists at $1090. Contact Teleray Division, Research Inc., P.O. Box 24064, Minneapolis, MN 55424 for more information, or call Jim Anderson, (612) 941-3300. *Circle No. 159.*

High Resolution 20" Monitor Added by CPT

A high resolution, high density **CRT display module** is available in OEM quantities from CPT Corporation. The module, HRD-20, uses a 20" CRT to display full pages of text or graphic data.

With a bandwidth of 105 megahertz, the non-interlaced system scans at 50,000 scan lines per second. Dot resolution is rated at 0.01 inch with clear definition; rise/fall time is less than 3 nanoseconds.

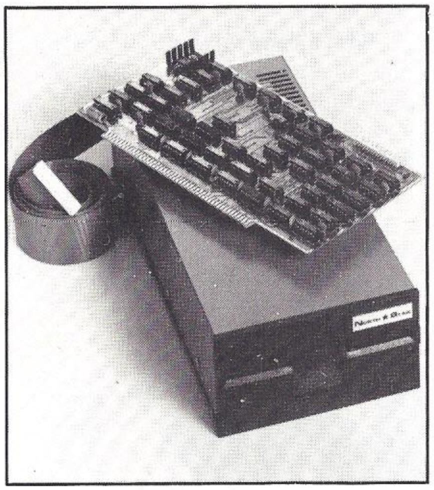
Geared for document retrieval, text processing and graphics applications, the high resolution screen is human en-

gineered for 8 hours per day use without eyestrain; the high scan rate and short rise/fall time yield a highly readable display, according to CPT.

For more information, contact CPT Corp., 1001 Second St. So., Hopkins, MN 55343; (612) 935-0381. *Circle No. 137.*

North Star Announces Double and Quad Density

North Star Computers, Inc., announced a new **controller board** that can record in double density on the Shugart SA-400L (single-sided, double density) and SA-450 (double-sided, double density) minifloppies. The con-



troller board formats the SA-400L for 180K bytes (ten 512 byte sectors on each of 35 tracks) and the SA-450 for 360K bytes (same as SA-400L, but with two sides). Thus a three-drive system with SA-450 minifloppies can access more than a megabyte on-line.

North Star's double and quadrupal density expands the information storage of the North Star Horizon computer and Micro Disk System. The Horizon is a Z80-based microcomputer system with built-in minifloppy disk memory. The Micro Disk System (MDS) is an S-100 bus minifloppy disk subsystem. Both Horizon and MDS are supplied with a version of North Star's extended disk BASIC.

The new controller board controls any combination of SA-400L and SA-450 drives in single density or double density. It can read diskettes previously written in single density. Also, it permits attachment of single-density SA-400L drives previously acquired for the original North Star systems.

North Star is upgrading its BASIC and DOS to accommodate the increased capacity and the performance of the SA-400L and SA-450 disk drives. Programs developed on the original single density North Star software will run in double density with little or no change.

A quad density (SA-450) version of the North Star Horizon and MDS system will be available in January, 1979, the company said. This quad version will be offered for $200 additional or $1799 for Horizon-1 kit and $899 for MDS-A kit. Additional drives for quad capacity will be $499.

For more information, contact North Star Computers, Inc.,2547 Ninth Street, Berkeley, CA 94710; (415) 549-0858. *Circle No. 154.*

Diablo Develops 96-Character Metal Wheel, New Type Fonts

Diablo Systems, Inc., will soon market a 96-character metallized daisy wheel which will allow the company's 1650 series terminals to perform high-quality, remote word processing functions.

The new wheel adds eight characters to the conventional 88-character set metal wheel to accommodate ASCII serial communications, enabling the terminal to perform word processing functions on-line to a large mainframe in a time-sharing system.

Firmware changes in the terminal permit direct spoke addressing and variable hammer intensity.

According to Diablo, metal wheels provide a sharper impression with better character definition, resist wear and have a longer lifetime than plastic wheels.

For more information, contact Diablo Systems, 24500 Industrial Blvd., Hayward, CA 94545. *Circle No. 141.*

New H8 Floppy Disk Kit from Heath

Heath Company announced their **kit floppy**, the H17. The H17 kit version is identical to Heath's assembled WH17 Floppy: 102K bytes of available storage area/disk, a fully-assembled Wangco Model 82 disk drive (expandable to dual disk), the interface/disk controller circuit board kit which plugs directly into the H8 mainframe and a self-contained power supply. The storage media is the expanded 40-track seek time and a typical random sector access time of less than 250 milliseconds for the new unit.

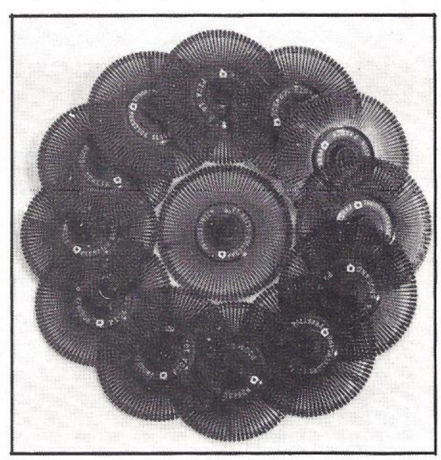
The operating system software for the H8/H17 Floppy Disk System designated H8-17, includes the Heath Disk Operating System (HDOS) with diagnostic for unit evaluation and optimization; the BUG-8 console debugger; TED-8 text editor; HASL-8 assembly language; and extended Benton Harbor BASIC. An extra diskette is also included.

For more information on the H17, the H17-1 (its optional second drive) and the H8-17 operating system software (priced at $530, $295 and $100 respectively,) send for a free copy of the latest Heathkit catalog. Write Heath Company, Dept. 350-680, Benton Harbor, MI 49022. *Circle No. 148*

Qume Adds 13 Printwheels for European Language Use

Qume Corporation has added 13 new **printwheels** for European language applications to the selection of printwheels for its daisywheel printers and data terminals.

The 13 printwheels consist of six pica 10 pitch and seven prestige elite 12 pitch in seven language classifications. The language sequences are: Severige/Suomi (Sweden/Finland); Danmark/Norge (Denmark/Norway); France; Schweiz (Switzerland); Italia (Italy); Nederland (Netherland) and Espana (Spain). Each style has a full



96 character set including punctuation, accent marks and numerals.

For more information, contact Qume Corp., 2323 Industrial Parkway West, Hayward, CA 94545; Elliot Wassarman, (415) 783-6100. *Circle No. 158.*

Up Your North Star System to 600K Bytes

The H&H Texan expands your North Star **floppy disk capacity** from 90K bytes to 600K bytes with total software compatibility.

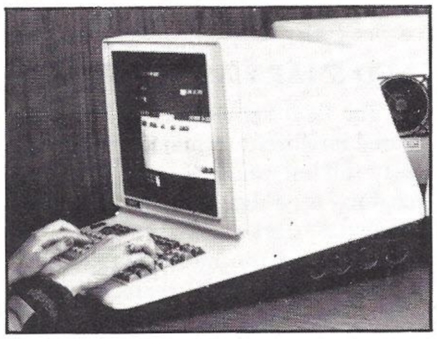
With the H&H Texan you can create a file for those 3000 accounts receivable invoices or 80 pages of text for your word processing applications.

The H&H Texan is available in three models; a single disk with 197K bytes for $595; dual drive with 394K bytes for $995; and three drives with 591K bytes for $1420. For more information, contact General Electronic Marketing 7315 Ashcroft, Suite 110, Houston TX 77801; (713) 772-9893. *Circle No. 147.*

New Omron 8038/UET CRT Terminal

A new **CRT computer terminal** from Information Products Division of Omron Electronics, Inc., provides a direct-replacement or add-on alternative for Uniscope 100 or 200 terminals, with added keyboard functions, a wider variety of display attributes, more versatile printer interfacing plus internal diagnostics and other features not available in its Uniscope counterparts. The new unit, Omron Model 8038/UET, is base-priced at $3200.

Omron's UET emulates Uniscope



line protocol, both synchronous and asynchronous, with comparable baud rates up to 9600. It meets Uniscope 100/200 hardware and software requirements, and provides all special Uniscope display characters. Essential Uniscope communications addresses – **RID**, **SID** and **DID** – are programmed via internal **DIP** switches, but may be overidden through the UET keyboard to allow terminal relocation or other system changes.

UET eliminates the need for external controllers. The unit plugs directly into existing multiplexers, or can connect directly to UNIVAC host processors via modem eliminator. Remote Omron terminals connect through modem or multiplexer in place of Uniscopes.

Keyboard functions provided as standard features in the Omron unit include ten function keys, forward and backward tabs, protected format and plus automatic delayed repeat that allows all keys to repeat at 15 cps after 1/2-second hold-down.

Omron UET provides eight levels of video field attributes for all characters: normal and reverse video; normal and reverse blink; dim and reverse dim; underline and video off. A 15-inch diagonal display presents 1920 characters in 24 lines of 80 characters each.

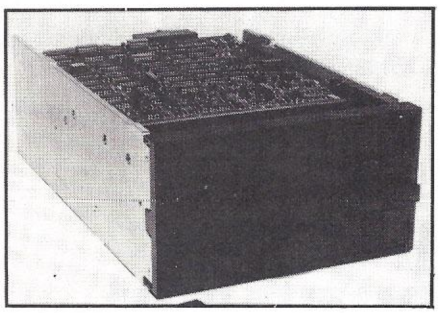
Internal diagnostic provisions, activated automatically upon power-up, review memory and keyboard integrity and advise the user of errors. Line Monitor Mode displays the entire communications protocol.

Auto Answer Mode automatically shifts the terminal into remote whenever host communications appear on line, then automatically reverts the terminal to local mode when host messages end.

For more information, contact Dan Daugherty, Omron/IPD, 432 Toyama Drive, Sunnyvale, CA 94086; (408) 734-8400. *Circle No. 155.*

Pertec Announces FD650 Floppy Disk Drive

Pertec Computer Corporation announced the FD650, a standard-sized (8-inch) **double-headed flexible disk drive** capable of recording and reading



data on both sides of an IBM (or equivalent) Diskette 2 or 2D. The drive also features double density recording to offer an immediately addressable unformatted storage capacity of 1.6 Megabytes.

The FD650 will be marketed on an OEM basis by Pertec Division and will also be incorporated into several MITS/Altair and iCOM systems.

The two-sided diskette drive is capable of double-density operation using MFM encoding. Each drive is equipped with a door lock, and the drive features die-cast construction. All electronics are on a single printed circuit board. Track-to-track access time is 3 msec., with head load time of 35 msec. and track settling time of 15 msec. The user can daisy chain up to eight drives.

Head positioning is achieved by a steel band attached to the head carriage and to a drive pulley on the shaft of a four-phase 1.8° permanent magnet stepper motor. Each step of the motor causes the head to move one track. The drive is housed in a die-cast aluminum base with a molded plastic bezel and electrically locked door.

The FD650 is priced at $755 in single quantities. For more information, contact PCC, Pertec Division, 9600 Irondale Avenue, Chatsworth, CA 91311; (213) 822-9222. *Circle No. 157.*

ECD SMART ASCII CRT

The ECD SMART ASCII is a CRT-based **intelligent terminal** that can display full line-printer format (up to 132 characters per line) and up to 40 lines on its 15" CRT with up to 4096 characters. Standard font is the full upper and lower case ASCII character set, but by using the supplied font editor program, the user can design his own special characters. The keyboard is relegendable so the user can easily modify it to match a new character set. Foreign language fonts can be implemented.

The interface for the SMART ASCII does not require any special protocol from the host. It communicates via a RS-232 line and looks like a simple printer/keyboard combination to the host, allowing either direct hookup or remote use via dial-up lines with keyboard selectable baud rates from 100 to 9600.

SMART ASCII comes with a sophisticated text editing program that allows complete off-line editing and supports transmitting data at a character, line or block at a time to the host. The SMART ASCII will also execute user written BASIC programs.

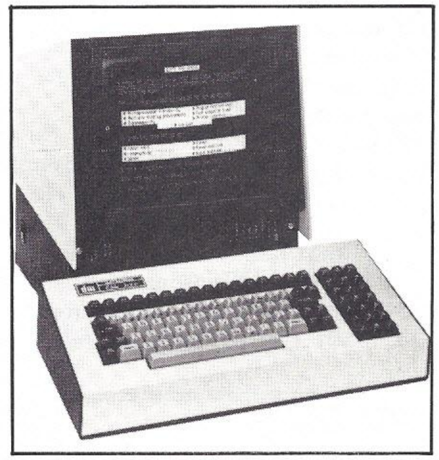
The system consists of a control unit with 37K of memory, a 78-key keyboard, a 15" CRT and 2 mini cassette drives at a price of $7900. For more information, contact Richard Eckhardt, ECD Corp., 196 Broadway, Cambridge, MA 02139; (617) 661-440. *Circle No. 143.*

Delta-1 Offers First Integrated Disk/Tape Capability

Meca's double-density **floppy disk storage system**, the Delta-1 provides up to 200K bytes of storage on a single 5-1/4" drive.

Included with the Delta-1 disk system is the MFM S-100 Disk Controller, which supports up to three SA-400 disk drives. Meca customers who own an Alpha-1 Tape System can use the MFM Disk Controller to combine the Alpha-1 and Delta-1 into a fully integrated tape and disk storage system.

North Star owners may use the MFM Disk Controller card to double



disk storage space from 90K bytes to 180K. The price for the controller card alone is $199.

Available software includes a CP/M disk operating system with editor, assembler, debugger and BASIC-E for $98. Microsoft Extended Disk BASIC is offered for $195. Several applications programs are available which operate with both the Delta-1 and the Alpha-1.

A special introductory price of $699 includes the mini-floppy single-sided disk drive, MFM Disk Controller, power supply, connectors and cable, documentation, and Meca disk operating system. For details, contact Meca, 7026 O.W.S. Road, Yucca Valley, CA 92284; (714) 365-7686. *Circle No. 152.*

Datamedia Buffered Video Terminal

Offering protected formats, video enhancements and APL overstrike/ASCII underscore, Datamedia Corporation's Elite 3045A is a low-cost, microprocessor-based fully buffered APL/ASCII **video terminal** with transaction processing capability. It features: character interactive, line or page mode communications; asynchronous and optional isochronous communications interfaces; 103 and 202 modem compatibility (201 modem compatibility with the isochronous interface) and switch-selectable EIA and optional 20 mA current loop interfaces; formatted data entry with protect capability; direct connect through RS-232C or

20 mA loop or remote connection compatible with Bell 103 or 202 modems; cursor addressability and remote position sensing; ten user function keys; no memory address space required to support screen enhancements; detached keyboard; fifteen baud rates, up to 9600, selectable from keyboard.

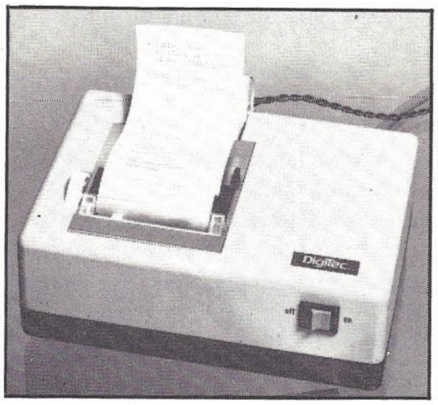
The Datamedia Elite 3045A is priced at $1995 in single quantities, and $1520 in quantities of 100. For more information, contact Datamedia Corporation, 7300 N. Crescent Blvd., Pennsauken, NJ 08110; (609) 665-2382. *Circle No. 140.*

Compact, Low-Cost Alphanumeric Printers

The DigiTec 6410 and 6420, small **desk-top printers** from United Systems Corporation, print 20 columns of alphanumeric characters first-line-up. They can replace bulky, expensive TTY terminals in applications that don't need 80-column capability, the company said.

An internal microprocessor makes these printers easy to interface. The Model 6410 provides a serial interface to RS-232C and 20 mA current loop systems at 100 baud. The Model 6420 works with 8-bit parallel bus systems at up to 1000 characters/second. They both respond to ASCII input.

The DigiTec 6400 Series alphanumeric printer produces fade-free records



by a quiet, electric writing technique. The high-constrat printout is easy to read. In addition, they can print bold face characters that don't lose their special emphasis when photocopied. Alphanumeric capability lets you record data and messages in a language that you and your customers understand. Sixty-four different characters (letters, numbers and symbols) are each produced in a 5 x 7 dot matrix.

In OEM quantities, USC will supply Model 6410/6420 printers in special colors and with your own identification or logo. These small, desk-top printers are for home and office. Price is $295 in 100 quantities; $395 in 1-9 quantities with off-the-shelf delivery. For more information, contact Gary Day, United Systems Corp., 918 Woodley Road, Dayton, OH 45403; (513) 254-6259. *Circle No. 258.*

The Writehandler, One Hand Typing Keyboards

A one handed keyboard for computers, terminals, displays and other 128 character ASCII or ISO coded devices is now available in both right and left hand configurations and in large and small sizes, featuring snap-action

switches, improved circuitry and Key-pressed signals as well as Strobe pulses to signal that data are available.

The small keyboard has wide application for touch typing and data entry where a free hand is needed, (telephone orders, computer programming, astronomy and microscope observations).

Connected to portable equipment, the Writehander allows inventory or field survey data to be entered with one hand. The Writehander can be interfaced with any computer, terminal, printer or other device that will accept parallel 7-bit code signals and provide the nominal power required.

To use the Writehander, the typist places his fingers on four finger keys and his thumb on one of four thumb key pairs. The four finger keys operate as the lower four bits of the seven bit ASCII code, and each thumb key generates a unique combination of the upper three bits. Red thumb keys are used for the lower case alphabet, blue keys for upper case, gray for digits, and so forth. Logical coding minimizes thumb motion and simplifies learning the codes.

Delivery is from stock to 30 days.

For more information, contact Sid Owen, NewO Co., 246 Walter Hays Drive, Palo Alto CA 94303; (415) 321-7979. *Circle No. 153.*

Threshold 600 Multi-Function Voice Data Entry Terminal

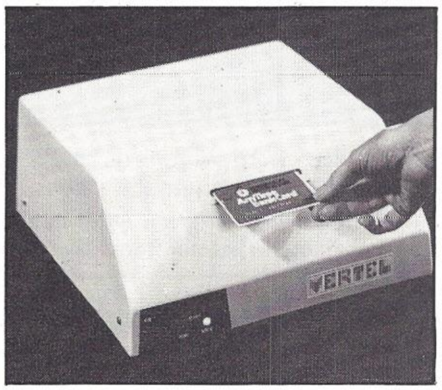
New literature describing the Threshold Technology 600 teletype-compatible **voice data entry terminal** is available from the company. The Threshold 600 enables an individual at the source level to communicate directly with a computer by voice and without the need of a special vocabulary, the company said. It can be a direct replacement for either a video or teleprinter terminal with no modification of host-computer software. The Threshold 600 controls voice recognition, speaker training, reference data storage and other voice recognition functions. Compatible with EIA RS232C, CCITT-V24, or 20 mA current loop teleprinters, it can be used in data entry, inquiry-response, timesharing, editing and command and control functions.

Standard features include a 16-digit alphanumeric display, dynamic vocabulary selection, ASCII coded output, local operator-controlled console and full duplex communications. Designed to operate in a working environment, the unit maintains its accuracy when the ambient temperature is between 41 degrees F and 104 degrees F and the humidity is between 10 and 90 percent, noncondensing.

Contact Threshold Technology, 1829 Underwood Blvd., Delran, NJ 08075; (609) 461-9200. *Circle No. 255.*

Low Cost Credit Card Reader

Vertel, Inc., today announced the firm's new ANSI-standard Series RW-31 Creditloader **credit card reader/writer**, which has been field proven in indoor



and outdoor bank terminal, security access control and petroleum applications. It's also suitable for POS terminal, personal identification and automated bank teller markets, the company said.

Credit cards may be read, edited and written with the RW-31 when connected to a minicomputer or intelligent terminal. The product may also be used with a dumb terminal as a low cost on-site credit card reader. With memory and software options soon to be announced, the RW-31 will be user programmable to function intelligently with the dumbest of terminals.

Standard RW-31 Thrift configuration provides read-only operation on track 2 (ABA) and read/write on track 3 (Thrift). Other single and dual channel head options are available. RW-31 electronics employ an F8 microprocessor with ROM and RAM, full I/O buffering and solid state motor control. ROM-changeable interfaces include byte parallel, RS232 with baud rates from 300 to 4800, and TTY. A power supply is also available, as is a free-standing desktop enclosure which can house all electronics and power supply.

RW-31 features accurate, "jitter-free" (plus or minus 4% max) data independent of head speed – a result of its enclosed, precision tracking magnetic head which travels along a precision, shielded ball bearing lead screw. A strobe controlled PC/LED/Shaft encoder precisely determines magnetic head lateral displacement by measuring the rotational displacement of the lead screw. The head design minimizes errors, data loss, damage or jamming due to card warpage, thickness variations or foreign materials and provides 500,000 or more "fail safe" operations even with unskilled operators. In addition, it is not necessary for the operator to "turn loose" of the card at any time, the company said.

Series RW-31 performance includes recording density of 150 or 420 bits/inch (59.1 or 165.4 bits/mm), an ISO/ANSI standard data block (track) length of 2.81" (71.4 MM), read/write cycle time of 1.4 seconds/stripe and plus or minus 4% speed stability (data jitter). The diecast RW-31 frame and chassis measures 8.187" L x 2.475" W x 3.200" H (20.79 cm x 6.3 cm x 4.1 cm) and weighs 30 ounces (0.85 kilograms). Power required is 12 VDC and 5 VDC. For more information, contact August A. (Gus) Toda, president, Vertel, Inc., 125 Ellsworth Street, Clifton, NJ 07012; (201) 472-1331. *Circle No. 259.*

New Tri-Data Flexifile 21

Tri-Data has announced their new FlexiFile 21 random-access, microprocessor-controlled, **flexible disk system for timeshare users.** The FlexiFile 21, a low-cost, high performance system, adds intelligence, storage and communications to CRTs and teleprinters. Combining the FlexiFile 21 with an acoustic coupler and a dumb terminal gives a smart, communicating terminal for timesharing applications.

Timeshare users plug their terminal and modem into the FlexiFile 21. Data entry and editing are performed off-line to reduce line charges and CPU access time. FlexiFile 21 disk storage provides users with an easy means of local data storage and filing and minimizes their remote computer data storage costs, according to the company. The Flexi-

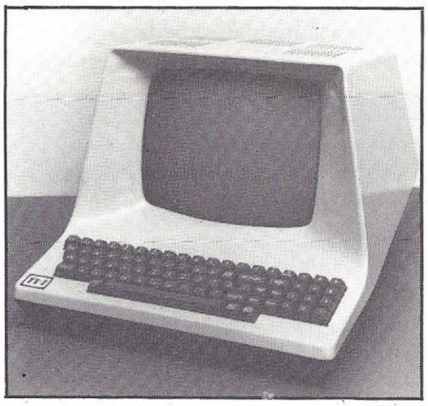
File 21 adapts easily from one timeshare system to another for high speed transmission of data, and it also provides an extended memory buffer for smart timeshare terminals.

Six front panel mode select switches enable the operator to select the correct operating mode status. Program functions are defined by Tri-Data's standard software package or as programmed to meet specific timeshare user requirements. Changing or storing programs is achieved in a few seconds by inserting a flexible disk and loading an existing or a new program from disk to RAM or a program from memory to disk.

The 8K random access memory is expandable to 12K. The FlexiFile 21 also incorporates two built-in, self-test features to verify that the read/write functions and memory are operating properly. The unit weighs 17 pounds and measures 17" x 6.6" x 8". Unit price for the FlexiFile 21 with a standard interface and 8K RAM is $1995. For more information, contact Jennifer Cherney, Tri-Data, 800 Maude Ave., Mountain View, CA 94043; (415) 969-3700. *Circle No. 256.*

Televideo Introduces Low Cost Computer Terminal

Televideo, Inc., has announced its new **smart computer terminal**, the TVI-912. According to Televideo, the com-



pact terminal features state-of-the-art design improvements which allow it to offer high level performance at low cost. The TVI-912 is compatible with all computers. Single unit price is $749 with quantity discounts available.

Standard features include an editing capability, protected field, addressable cursor, microprocessor control, line and character insert/delete, upper and lower case characters and tabbing. Switchable transmission rates ranging from 75 to 19,200 baud are also standard. Other features include printer port for hard copy and 2000-character additional second page memory.

For more information, contact: K. Philip Hwang, president, Telvideo, Inc., 3190 Coronado Drive, Santa Clara, CA 95051; (408) 246-5428. *Circle No. 160.*

Epson Model 10 Line Printer

Epson has introduced an upgraded microprocessor-controlled model 10 **Line Printer.** Either a parallel interface or a serial interface can be installed inside the basic printer cabinet. The OEM has the option of ordering the

serial interface without an inconvenient external serial interface unit or the unused parallel interface. Because space for an interface board is provided inside the basic cabinet, OEMs also can design their own interface if they desire.

A switch activated, self-printing function commands the printer to perform a pre-programmed print exercise, without being connected to the computer or an external printer exerciser. This feature lets the user see if a problem exists, and if it does, to determine if it is in the printer or the computer.

EBCDIC code (48 character set) can be accepted by the model 10. The minimum mechanical printing rate of 150 lpm is not changed even if the 48 character set is aligned on the belt; however, 16 out of 48 characters can be printed at the rate of 176 lpm.

Model 10 with a 96 character set (upper and lower cases) ASCII has a printing rate of 84 lpm; 32 out of 96 characters can be printed at the rate of 150 lpm.

OEMs can specify 64 ASCII or 96 ASCII in addition to 48 EBCDIC without extra charge, and friction feed or tractor feed with vertical format unit is available at extra cost.

For more information, contact Epson America, 23844 Hawthorne Blvd., Torrance, CA 90505; (213) 378-2220. *Circle No. 145.*

Discus I Full-Size Floppy Disk From Thinker Toys

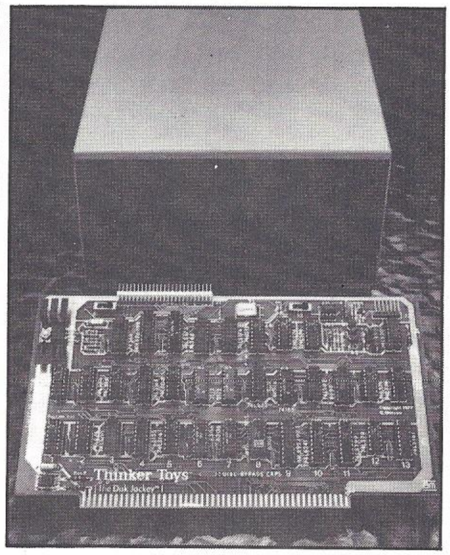
Morrow's Micro-Stuff/Thinker Toys is delivering the new Discus I full-size **floppy disk memory** for S-100/8080 microprocessor systems. The Discus I system comes completely assembled and tested, with all required hardware and software, for $995 (plus tax and handling).

Hardware in the Discus I system includes a Shugart 800R full-size disk drive mounted in a custom, all-metal cabinet with an independent power supply, a Disk Jockey I S-100 controller with a capacity for seven additional disk drives, and all necessary cables and connectors.

The controller offers an on-board serial I/O port to which all system software has been interfaced. This feature allows the system to be plugged in and brought up immediately. The I/O routines can then be modified with the included system software.

Software included in the Discus I base price features an integrated DISK/ATE system containing most utilities: Disk Operating System, File Management, System Debugger, Text Editor, Batch Processor and 8080 assembler.

Also included in the base price is BASIC-V, a virtual disk BASIC with the ability to address up to two megabytes and to accomodate a wide variety of data types including string-oriented



arrays with an unlimited number of dimensions. Also included are patches for CP/M.

For more information, contact Neila Richmond, Thinker Toys, 1201 10th St., Berkeley, CA 94710; (415) 524-5317. *Circle No. 254.*

E&L Instruments Adds VTE-1 Video Terminal

Used with a television set or monitor, the VTE-1 **Video Terminal Electronics system** provides a full ASCII keyboard, reprogrammable character generator, cursor and flicker-free refresh. Full duplex and local operation are possible, with RS232C and 20 mA current loop interfaces operating at speeds of 75 to 9600 baud.

The standard character set contains 64 upper case alphanumeric ASCII characters with lower case optional. The character generator is user-programmable to provide any user-defined 5 x 7 dot matrix characters. The 128-character ASCII decoder supplied is also user-reprogrammable to accommodate alternate character codes or code sets.

The blinking full field cursor can be directly positioned in any screen location, or turned off for a clean graphic display. A nondestructive read screen function transmits the ASCII code for the character at the cursor position to the user, with automatic spacing. Other features include clear screen, bell code and wrap-around.

The VTE-1 provides a 75 ohm composite video output that will directly drive any U.S. compatible black and white television monitor, or the optional MON-1 monitor available from E&L. Standard TV sets may be connected through readily available VHF converters or modified by qualified servicemen to provide a direct video input connection.

List price of the VTE-1 is $600 fully assembled, $450 in kit form. Optional MON-1 monitor (assembled only) is priced at $210. For more information, contact E&L Instruments, Inc., 61 First St., Derby, CT 06418. *Circle No. 144.*

Futuredata's 1 Megabyte and 2 Megabyte Floppy Disk Units

Futuredata Computer Corporation has announced Universal **Microcomputer Development Systems** offering double-density and double-sided, double-density dual drive floppy disk units. Microsystem/31 is offered with double-density disk; Microsystem/32 features a double-sided, double-density disk.

Microsystem/31 may be ordered with 8080, 8085, Z-80, 6800 or 6802 microprocessor. The system comes with 16K bytes of RAM memory, CRT display, keyboard, dual drive 8" double-density floppy disk unit, DOS software on diskette and a full set of manuals. Each diskette has a formatted capacity of 512,512 bytes or 1,025,024 bytes for the dual drive system. Microsystem/32 is identical, except double-sided, double-density storage provides 1,025,024 bytes or 2,050,048 bytes per dual drive system.

Futuredata universal Microsystems provides hardware and software development capabilities. For hardware development, Microemulator In-Circuit Emulator and debug packages are available for all microprocessors in the Futuredata line. For software development, a set

of software packages including monitors, debuggers, editors, assemblers, utilities, relocatable macro assemblers, linkage editors and BASIC compilers is available for all supported processors.

All Futuredata Microsystems include the microprocessor CPU of your choice, with up to 64K bytes of memory, high speed 960 character CRT, ASCII keyboard, dual floppy disk or cassette tape unit, operating system software and documentation. Optional accessories and software include Microanalyzers, in-circuit emulators, line printers, BASIC compilers, RDOS (disk operating system with relocatable macro assembler and linkage editor), and word processors. Low cost plug-in modules permit the system to be converted to 8080, 8085, Z-80, 6800 or 6802 processors.

System features include two RS-232 serial ports, 8-bit parallel TTL I/O port, realtime clock, bootstrap in PROM, memory write-protect under software control, 8-level vectored interrupts, DMA capability and complete disk and tape operating systems.

Microsystem/31 with double-density floppy disk unit and processor costs $7975. Microsystem/32 with double-sided, double-density floppy disk unit and processor costs $8975. For more information, contact Futuredata Computer Corporation, 11205 South La Cienega Blvd., Los Angeles, CA 90045; (213) 641-7700. *Circle No. 146.*

Three New Data Communications Products

Three new additions to the USR-300 series of **data communications products** feature asynchronous, half/full duplex operation at data rates of up to 300 baud over voice-grade telephone lines and are compatible with Bell 103/113 style modems.

The USR-310 is an originate-only acoustic coupler that features an RS 232C interface and connects to phone lines via any standard telephone. Unit-quantity price is $139.

The other two new offerings are the FCC-certified configurations of the USR-320 and USR-330 modems. The units are now available as FCC-certified packages for direct connection to telephone lines. The phone line connection is via the standard modular style voice jack (USOC code RJ11C), used by most telephones. The USR-320, which features auto-answer-only operation, is priced at $299 in single piece quantities. The USR-330, which features front panel switch selection between manual-originate and automatic-answer operation, is priced at $324 in unit quantities. The units are still available for use with a telephone company provided CBS-1001F Data Access Arrangement. In this configuration, the USR-320 is priced at $160 while the USR-330 is priced at $185. Both units are available with RS232C and/or 20 mA current loop interfaces.

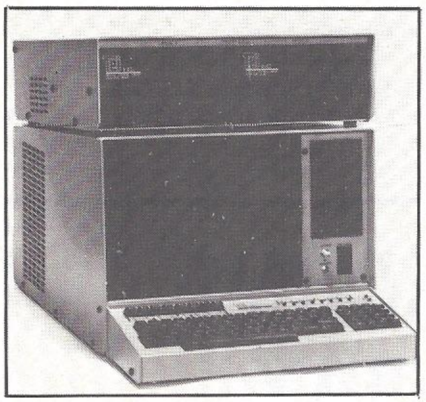
Contact U.S. Robotics, Inc., P.O. Box 5502, Chicago, IL 60680; (312) 528-9045. *Circle No. 257.*

SYSTEMS

Processor Terminal TEI PT 212/80

CMC Marketing Corp. has announced the availability of another model in the new Processor Terminal series. The TEI PT 212/80, is a complete and **self-contained computer system** with display, disk storage, a full keyboard and a 12-slot motherboard. It may be used either as a standalone processor or as a processor terminal.

Features of the PT212/80 include a 15" high-resolution monitor with a face plate of smokey plexiglass to reduce glare and enhance type visibility, a full upper and lower case ASCII keyboard with eight user designated special function keys and a 16-key numeric cluster



pad. Two Shugart SA-800 Standard floppy disk drives are included. The 12-slot mainframe contains a CPU board that features an 8080 processor and a special circuit that implements a start up "jump to" routine to any user selected byte address. Press the reset switch and the system boots to your preselected address. 32K static RAM memory is provided with additional RAM as an optional item. A disk controller which will handle four standard drives. The 8" disk media is soft sectored and has a capacity of about 250 KB. The video controller board uses a 24 x 80 format with many special features. The I/O board provides three parallel and three serial ports with selectable baud rates of 75 to 9600. Outputs are RS-232 and TTL.

The unit is housed in a heavy duty aluminun cabinet with power provided by a constant voltage transformer (CVT) power supply that makes brownouts a thing of the past, according to CMC. Fan, washable filter and a full complement of spare edge connectors for ancillary cards are included. Optional software available for use with the PT212/80 includes CP/M operating system, SuperBASIC, and Fortran. Cobol coming shortly. The Processor Terminal Model PT212/80 fully assembled and tested is priced at $5995. OEM and dealer pricing is available upon inquiry. Other models in the PT series include the PT112, PT312/80, PT412/80, PT208 and the PT 408/80. For more information contact CMC Marketing Corp., 5601 Bintliff, Suite 515, Houston, TX 77036; (713) 783-8880. *Circle No. 104.*

Sol Star 8080 Data Processing System

THE Sol Star 8080 Data Processing System is a turn-key package designed for small business applications, including word processing, bookkeeping, engineering, legal billing, apartment management, document preparation and data base management.

The model includes 32K of memory and a Multiterm/Diablo printer. Based on the popular SOL-20 Computer and North Star disk drive combination, Sol Star adds special software, dual-sided disk recording with Pertec drive units and custom cabinetry.

In the word processing mode, Sol Star provides fast bi-directional printing, storage capacity of approximately 100 typewritten pages on a single $4.50 diskette and convenient summary of operator instructions within the program. Telecommunicating versions are also available.

The Sol*Star soso DP System costs $1000. For more information, contact Orange County Computer Center, 1913 Harbor Blvd., Costa Mesa, CA 92627. *Circle No. 105.*

APF's PeCos I Personal Computer

APF Electronics, Inc., introduces PeCos I, a complete **Personal Computing system** incorporating comprehensive math capabilities, exceptionally large memory and ease of programming in the most English-like computer language ever devised.

PeCos I, short for PErsonal COmputing System, is a fully integrated computing system. It combines a 9" CRT, a standard size 60-key keyboard and dual cassette decks.

The easy-to-learn PeCos language makes it possible for almost anyone to use the computer without lengthy training in a complex language. PeCos language is a derivative of the JOSS language developed by Rand Corporation. Users have found that PeCos language is much easier to learn and program than BASIC, according to APF.

PeCos I's math program permits full computation in nine-digit floating decimal arithmetic with a number range from 1 x 10^{-99} to 1 x 10^{99}. PeCos I has built-in all the functions of a programmable calculator and more, including trigonometry, number dissection, string concatenation, transcendental and the ability to define functions.

PeCos I also has memory capabilities with 24K ROM and 16K RAM internal. PeCos I has built-in dual cassette decks that are semi-automatically controlled. The cassette decks use standard audio cassettes which can each store up to 80K bytes of information. It is possible to read from one tape and write to the other. All of PeCos I's I/O's are done at a baud rate of 800 speed tolerant recording.

PeCos I measures 18½" x 19½" x 8½" and also includes digital tape counters for rapid data retrieval, tape files addressable by either name or number and two optional cassette decks to further expand the system.

ReCos I requires no hook-up with RF adapters, TV's audio cassettes or other peripherals. The self-contained system is all that is needed to be up and

running in the home, office, laboratory or school. Everything required to operate is included standard: the 60-key, full-size keyboard with 110 codes and upper and lower case; the 9" CRT displaying 16 lines of 40 characters each

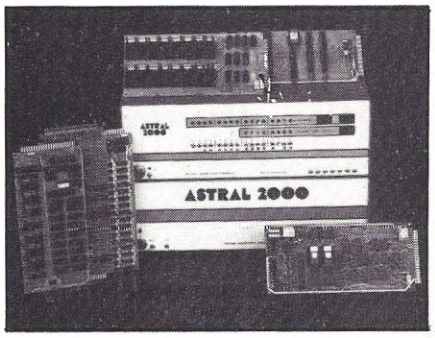
with automatic scrolling and speed control; built-in dual cassette decks; 6502 microprocessor; power supply; and RS 232 transmit port that lets a serial printer output 80 characters per line, with upper and lower case, if the printer has such ability.

PeCos I costs $1695. For more information, contact APF Electronics, Inc., 444 Madison Avenue, New York, New York 10022; 212/758-7550. *Circle No. 101.*

Astral 2000 Computer

A versatile microcomputer is available from Astral Computer designed for the OEM. The Astral 2000, based on the 6800 microprocessor, is available as a stand alone single board computer or in one of two enclosures complete with power supplies and 12 position mother board.

The Astral 2000 serves many uses because of its modular design. Separate cards contain the processor, memory, I/O and floppy disk interface. The system is fully supported with an extended 8K BASIC, Assembler, Text Editor and Disc Operating System Software (DOS). Two separate interchangeable front panels are also available. All



system cards are 10" by 4.5" to fit into limited space and have standard dual 22 pin edge connections.

For the OEM interested in maximum access to registers, interrupts and system diagnostics, Astral provides a full front panel enclosure complete with interrupt, register and data switches and hexadecimal and binary displays. This panel is especially useful for development work because of the access it provides to the system. A limited control front panel suitable for most applications includes power on reset, reset switch, user defined interrupt switch and power on key switch. The heart of the Astral 2000 is the Processor Board. This board contains the 6800 microprocessor, firmware and RS-232 or 20 ma TTY current loop interface circuitry. Reader control, clock and the necessary power outputs required to operate common peripherals are also provided at the pin connector. The serial I/O is isolated through optoelectronic devices. Operation through this port is at 110 or 300 baud.

A software selectable echo back capability is built into the processor board. Both address and data lines are fully buffered through tri-state devices for maximum flexibility and reliability. The processor provides DMA through "cycle stealing" which minimizes program execution delays. For connection to slow memory devices such as EPROMs, "Refresh Grant, Memory

Ready" and "Memory Clock" controls are available at the edge connector. A full set of microporcessor control lines including "Valid Memory Address, Ø2, Halt, Reset, Single Step" and "Tri-State Control" are brought out to the edge connector.

For more information, contact Astral Computer Co., 991 Commercial St., Suite 2, Palo Alto, CA 94303 (415) 494-8048. *Circle No.102.*

DTC Introduces New 'Taskmaster' Small Business Computer

Data Terminals and Communications (DTC) announce the TaskMaster, the company's first totally **integrated small business system.** Taskmaster provides small business powerful and easy-to-operate data and word processing capability.

The TaskMaster system consists of a flexible disk-based microcomputer, a full-page video screen, operator console, and a 45cps daisy wheel printer.

The system's hardware provides users with sophisticated microprocessor architecture, 64K memory (8K ROM, 56K user RAM). The unit is provided with a dual diskette drive. Space is provided for expansion to a second dual diskette.

The TaskMaster includes a proven general business software package, the Accountmaster, which performs a full range of accounting functions. The TaskMaster can also be expanded to include three optional DTC software packages: the Stationmaster (distributed processing), Textmaster (word processing) and Mailmaster (electronic mail.)

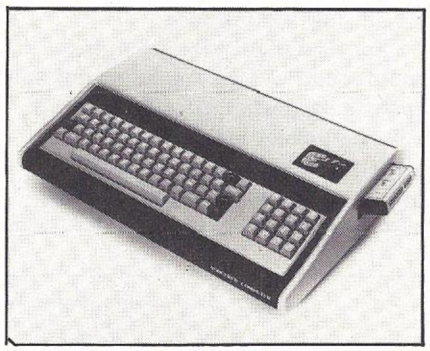
Other optional features include a hard disk for mass data storage up to ten megabytes (five MB fixed and five MB removable) and an optional matrix printer at 200 characters/sec. Price for the TaskMaster and standard accounting software package is less than $20,000. For more information, contact Data Terminals and Communications, 590 Division Street, Campbell, CA 95008. *Circle No. 106*

Exidy's New Personal Computer, The Sorcerer

Exidy Incorporated enters the consumer electronics market with its new user-programmable personal computer, the Sorcerer. The self-contained Sorcerer needs only to be plugged into a video display and a cassette tape recorder to be a fully-functioning computer system.

The Sorcerer has plug-in Rom Pac cartridges. The Rom Pacs contain high-level programming languages, operating systems or special proprietary software. Each Sorcerer comes with a Rom Pac cartridge containing Standard BASIC. Additional Rom Pacs available or now in development include a user-programmable EPROM Rom Pac, an assembler editor, a disc operating system and a word processing package.

Applications programs are stored on inexpensive magnetic tape cassettes and



can be loaded from one or two tape recorders through the Sorcerer's dual cassette interface at data rates of either 300 or 1200 baud.

The Sorcerer can be used as a smart terminal for communications and time-sharing applications. Its RS232 serial interface accepts a modem to transmit data through phone lines at 300 or 1200 baud.

The Sorcerer's molded case contains a professional data processing typewriter-style ASCII keyboard with 79 keys providing full upper and lower case alphanumeric characters and graphic symbols. A 16-key numeric pad speeds information entry and inquiry. The Sorcerer offers a total of 256 graphic expressions. In addition to its 128 ASCII set, it has 64 characters designated on the keyboard and a second set of 64 may be identified by the user definition. Alternatively, the two sets of 64 may be identified by the user through program control for full custom applications.

Extremely fine graphic resolution of 122,880 points on the video screen is produced in a 512 x 240 format for detailed illustration. Character output provides the largest capacity available in a personal computer. The Sorcerer displays a total of 1920 characters on the screen at one time in 30 lines by 64 characters in an 8x8 format. Hard copy of the Sorcerer's output can be produced by attaching a printer to the Sorcerer's parallel interface.

Exidy provides 12k of ROM memory in the Z-80 powered Sorcerer which includes a power-on monitor program and Standard BASIC. Because of the ROM PAC cartridge system, the Sorcerer requires less RAM memory, reducing the cost of computing, according to Exidy. The Sorcerer contains 8k of RAM for user program space and is expandable internally to 32k.

The Sorcerer's capacity and capabilities are expandable with the addition of Exidy's 8-slot S-100 bus module and the wide variety of available S-100 boards. Suggested retail price is $895 with a Standard BASIC Rom Pac cartridge. For more information, contact Paul Terrell, Marketing Manager, Data Products Division, 969 W. Maude Aven Avenue, Sunnyvale, CA 94086; (408) 736-2110. *Circle No. 107.*

Futuredata's System Supports 8086

Futuredata Computer Corporation announces the addition of the 8086 **16-bit microprocessor** to the array of 8-bit processors supported by its universal microcomputer development system.

Futuredata systems also support the 6802, 6800, Z-80, 8080 and 8085 processors. The 8086 systems are software-compatible with both the 8080 and 8085 but deliver up to ten times the processing power at 5MHz. The 8086 system provides direct addressability to a megabyte of memory, with both 8-bit and 16-bit signed or unsigned multiply and divide in hardware. In addition, the system provides more efficient byte-string operations, improved bit manipulation, dynamic relocation, re-entrant code, position-independent programs and instruction look-ahead. Assemblers, translators and optional compiler packages permit translation of existing 8080 and 8085 programs to run on the 8086.

The Futuredata 8080/8085 to 8086 translator utility, which is included with the standard system software, permits direct assembly language conver-

sion of programs. Source programs written in 8080/8085 assembly language can be translated to the new 8086 assembly language.

Futuredata universal microsystems are complete microprocessor development tools providing both hardware and software development capabilities. For hardware development, microemulator In-Circuit Emulator and debug packages are available for all processors. For software development, a full set of software packages including monitors, debuggers, editors, assemblers, utilities, relocate macro assemblers, linkage editors and BASIC compilers are offered for all processors.

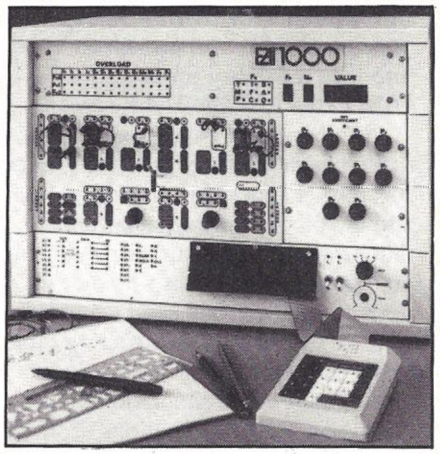
A full range of disk-based 8086 systems are offered to allow the designer to select the level of performance required. Each includes microprocessor CPU with up to 256K bytes of memory, high speed 960 character CRT, ASCIL keyboard, dual floppy disk unit, operating system software and documentation. Optional accessories and software include Microanalyzer logic analyers, in-circuit emulators, line printers, BASIC compilers and RDOS (disk operating system with relocatable macro assembler and linkage editor). Low cost plug-in modules convert the systems to any of the other processors.

Microsystem/32 MOD 8086, including 32K bytes of memory and dual 8" double-sided, double density floppy disk unit costs $9625. For more information contact Futuredata Computer Corporation, 11205 South La Cienega Blvd., Los Angeles, CA 90045; (213) 641-7700. *Circle 109.*

Electronic Associates Introduces Desk-Top Analog Hybrid Computer

Electronic Associates, Inc. introduces a new **desk-top analog/hybrid computer** for education and research. The new EAI 1000 system features a keyboard-directed microprocessor control system, removable patch panels, universal trunking, individual analog component monitoring and modular construction.

The new computer is available in three basic configurations ranging from a 4-integrator, 6-summer, 2-multiplier system to a 12-integrator, 18-summer,



6-multiplier system. Prices range from approximately $4900 to $12,000. Provision for two to six special functions such as sine-cosine and vector genera-

tors provide a balanced computation system.

Basic digital logic built into the analog section of the EAI 1000 supports control for many applications. A digital logic expansion module, with counters and clocks, is available for more extensive requirements. This expansion, priced at $1250, also provides a hybrid interface capability for using the EAI 1000 in the EAI MiniHybrid System.

A full range of peripheral devices including an arbitrary function generator, four-channel display oscilloscope and x-y plotter are available for handling empirical data and displaying program results. For use as a laboratory device, the EAI 1000 can be purchased with an external trunk buffering capability to scale instrument signals for direct input into a simulation model or signal conditioning functions of the computer program.

For more information, contact Electronic Associates, Inc., West Long Branch, NJ; (201) 299-1100. *Circle No. 108.*

Burroughs Introduces Modular Transaction Control Systems

Burroughs Corporation has introduced a new family of on-line, **modular transaction control systems.** They include on-line data communications control processors, plus modular keyboards, displays, printers and special purpose devices which can be intermixed to fit a variety of requirements in financial institutions, government offices, school systems, hospitals, sales offices, plants and warehouses.

Versatility and economy are primary features of these new modular systems; the new systems are compatible with Burroughs on-line terminals and transaction control units already installed, and may be used in conjunction with them.

The modular components include TD 500 input and display systems, AP 100 and AP 300 printer terminals, and DC 128/129 systems and communications processors.

The TD 500 is a new series of input and display modules featuring a choice of 5" or 9" cathode ray tube screens. Modular options include a numeric keyboard, an alphanumeric keyboard, a personal identification number (PIN) keyboard and a magnetic card reader. These components can form one compact unit, or be separated for convenient placement on work stations or financial teller counters where space is limited. The PIN keyboard and magnetic card reader provide privacy for bank customers.

Each component is priced separately. Purchase price for a TD 500 system with a 5" display screen plus a numeric keyboard is $1975. The same two modules are available for $84 per month on a one-year lease. A TD 500 with a 9" screen, plus an alphanumeric keyboard, PIN keyboard and magnetic card reader may be purchased for $2985, or leased for $128 per month. TD 500 systems will be available in the first quarter of 1979.

The AP 100 is a new series of auxiliary printers for use in financial institutions. The AP 100 offers all the forms-handling features of Burroughs' TU 1800 on-line teller terminals, including automatic passbook reading. A single AP 100 printer can serve two or more tellers. In other business settings, the AP 100 provides the economy of a shared printer for applications such as validating, receipting and disbursing in which documents are inserted and posted.

Purchase prices for the AP 100 start at $5150. Lease price is $196 per month for the basic AP 100. Deliveries are scheduled for the second quarter of 1978.

The AP 300 is an entirely new series of compact, fast, low-cost printer stations. An AP 300 can function as an unattended, desktop message printer, or as a printing, validating and receipting station. Controlled by a microprocessor, the AP 300 can print in multiple sizes and styles of characters, allowing reports, customer statements and other records to be customized for better readability or for drawing attention to particular areas.

Under control of a central computer, use of an AP 300 can be shared by several persons who are using Burroughs input keyboards and/or display screen terminals. The basic AP 300 printer can be purchased for $1700, or leased for $67 per month. These terminals will be available in the fourth quarter of 1978.

Burroughs DC 128/129 Series of systems and communications processors function as local control centers for the new transaction control modules and certain other teller terminals, transaction terminals and displays in a local network.

The independent DC 128/129 processors relieve the user's central system of the principal administrative burden associated with on-line data communications. Additionally, a DC 128 or 129 enables its own network to operate off-line. This means that customer services can continue despite a telephone line failure or a malfunction at the central computer site.

DC 128/129 systems occupy less floor space than a small file cabinet and, because they operate unattended and without special air-conditioning, they can be placed almost anywhere in the branch office.

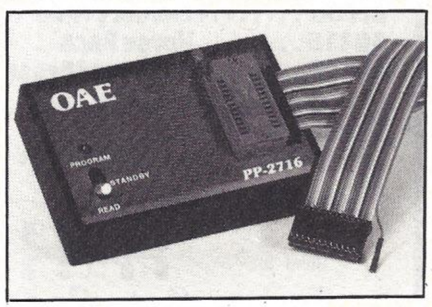
The basic DC 128 with magnetic tape cassette auxiliary storage may be purchased for $5150, or leased for $185 per month. Deliveries are scheduled for the first quarter of 1979. A DC 129 with full user memory, mini-disk, auxiliary storage, and a full data communications capability is priced at $6700 for purchase, or $240 per month on lease. Deliveries will begin in the second quarter of 1979.

For more information, contact Neil T. Jackson Corporation, Detroit, MI 48232; (313) 972-7269. *Circle No. 103*

TEST EQUIPMENT

Oliver Advanced Engineering has PP-2716 PROM Programmer

Oliver Advanced Engineering, Inc., announces a new **programmer** in their PP-Series: the PP-2716 for the single supply Intel 2716 EPROM. A 5-foot flat ribbon cable connects this programmer to any read only PROM sock-



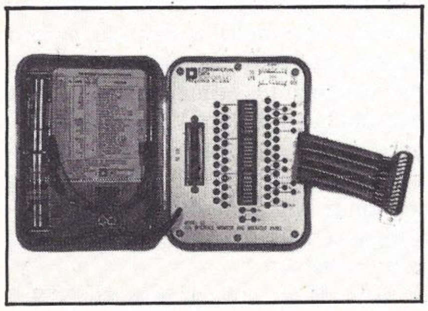
et via a 24-pin plug. Using OAE's PROM socket interface, data is sent over the eight lower address lines to

the programmer. No additional power supplies are required, and all timing and control sequences are handled by the programmer.

Each programmer comes with an internal DC to DC switching regulator and zero insertion force socket. The unit is packaged in an anodized aluminum case for table top operation. Each unit comes completely assembled, tested, aligned and ready for use for $295. Contact OAE, Inc., 676 West Wilson Ave., Glendale, CA 91203; (213) 240-0080. *Circle No. 195.*

Model 60 EIA Interface Monitor and Breakout Panel

International Data Sciences, Inc., has introduced a rechargeable version of their Model 60 EIA RS-232-C **Interface Monitor and Breakout Panel.** The Model 60 is a pocket-sized, portable test set providing access to all 25 conductors of the EIA RS-232-C interface. Twenty-four switches allow all interface conductors (except frame ground) to be individually interrupted for simulating "handshaking" control signals and isolating terminal and modem signals. Twenty-five test points on each side of the breakout panel provide access to EIA signals for test equipment probes. Twelve LEDs display



twelve key signals without interruption of signal path. Two additional LEDs sense whether signal levels meet EIA specifications. Small jumper cables allow cross-patching and monitoring of signals.

LEDs and test points are arranged in a natural format with respect to signal direction. Model 60 comes in a durable plastic case with hinged cover, weighs 13 ounces and measures 3-3/4" W x 5" H x 1-3/4" D. The unit operates directly from 110 VAC while recharging its two AA nickel-cadmium batteries.

Model 60 is priced at $270. Delivery is 30 days ARO. For more information contact Marketing Department, International Data Sciences, Inc., 100 Nashua Street, Providence, RI 02904; (401) 274-5100. *Circle No. 199.*

EICO Adds New Model 4A4 Multimeter to Truvohm Line

EICO Electronic Instrument Co., Inc., offers a new **multimeter**, Model 4A4, in its Truvohm line of low-cost, factory-assembled VOMs. The multimeter is a 4000 ohms/volt general-purpose instrument with 17 ranges. The meter housed in a high-impact plastic case measures up to 1000 volts DC and AC, up to 250 mA DC and up to 2 megohms resistance. Accuracy is plus or minus 3 percent on DC and plus or

minus 4 percent on AC. The unit features a recessed selector switch and a 3-inch meter with mirror-back scale. The movement is diode-protected. Suggested price is $17.95 assembled with batteries and test leads.

The EICO Truvohm line includes a 100K ohms/volt bench-size multimeter and a clamp-on AC current tester. The line also includes seven other instruments ranging from a simple, inexpensive 1000 ohms/volt meter to a 20,000 ohms/volt mirrored-scale meter. For more information, contact EICO Electronic Instrument Co., Inc., 108 South Road, Hicksville, NY 11801; (516) 681-9300. *Circle No. 191.*

CSC Experimentor Line Expands with EXP-325

The sixth and smallest addition to the Continental Specialties Corporation Experimentor series of **solderless breadboard sockets** is their Model EXP-325 featuring DIP-standard 0.3" center channel spacing. The new EXP-325 measures 1.86" long by 2.1" wide by 3/8" thick, about the size of a small cookie; yet it offers two rows of 11 five-point terminals each, plus two ten-point bus strips.

The front-access recessed screw-mount feature molded into other units in the Experimentor series has been omitted from the EXP-325 because of its compact size; however, the socket includes in its mold design a capability for rear mounting with 6-32F self-tapping screws.

Like other boards in the Experimentor series, the EXP-325 features CSC's tongue-and-groove edge construction, which also permits attaching a small "front panel" plastic or metal sheet to hold controls, indicators or other oversize components.

Suggested price in unit quantities for the EXP-325 is $2.75. For more information, contact Continental Specialties Corporation, 70 Fulton Terrace, New Haven, CT 06509; (203) 624-3103. *Circle No. 192.*

World's Smallest 12 Volt, 120 mA Modular Supply

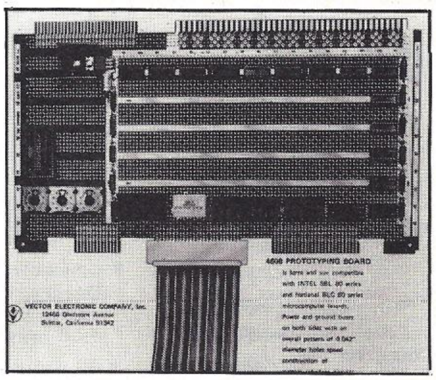
Calex recently announced the 21-12 power supply, another in the series of **miniature power supply modules** for MOS devices, relays or lamps.

The 21-12 is 1.75" wide by 2.25" long, stands 1" high and weighs 150 gms (5.3 oz). The transformer is vacuum impregnated and carries a 5-year warranty. Output voltage is 12 volts plus or minus 1% at 120 mA. Line and load regulation are plus or minus 0.1% and noise and ripple are less than 2 mVrms. The model is short circuit protected with fold back current limiting. It's designed for printed circuit mounting and has a threaded insert molded into the case to mechanically secure it in place. Input power is 115 VAC plus or minus 10% at 50-400 Hz. Inputs of 100 VAC and 220 VAC at 50 Hz are also available.

The Model 21-12 is priced at $29 each in single quantities and $19 each at 100 pieces. For more information call Ron Kreps at (415) 932-3911 or write to Calex Mfg. Co., 3355 Vincent Road, Pleasant Hill, CA 94523. *Circle 193.*

Low-Cost Prototyping Boards Aid Series/80 Microcomputer Design

Two new **prototyping boards**, form and size compatible with either Intel SBC 80/10and SBC 80/20 or National BLC-80/10, BLC-80/11, BLC-80/12 and BLC-80/14 microcomputer boards, have five double-sided card-edge con-



nectors for convenient bus-oriented, parallel or serial input/output. Available from Vector Electronic Company as models 4608 and 4608-1, the 12" by 6.75" by 0.042" boards are pre-punched with 0.042" diameter holes on 0.1" grids. They accommodate DIPs with 0.3", 0.4", 0.6" and 0.9" lead spacing as well as solid-state relays, electro-mechanical relays and thumbwheel switches.

The 4608 model has power and ground buses to reduce wiring labor and improve performance on custom interface boards. Pad arrays surrounding three-hole groups, permit soldering DIPs and interconnections or easy solder mounting of wrap-post DIP sockets. Board intra-connections are made with wrapped wire or solder. The Model 4608 holds up to 54 16-pin DIPs in the patterned area and also has a 13-square-inch unclad area for freedom in component placement.

Heavy-duty power and ground buses on the component side and a dual network of power and ground buses on the wiring side route power from the system bus connectors or from the pads of a TO-220 size regulator to areas beneath the DIPs. Card-edge connector contacts are unassigned, except for the four end contacts on the 86-terminal connectors which are connected to the ground bus. This feature allows the designer full discretion in selecting any of the 12 designated Series-80 power supply lines for routing on the board. The buses also may be cut into segments to accomodate different voltage levels in different board areas.

Vector's Model 4608-1 is identical to Model 4608 in size and edge connectors, but has no etched pattern. With continuous array of 0.042" diameter holes on a 0.1" grid, it holds up to 144 16-pin DIPs.

The five card-edge connectors are compatible with Intel's Multibus or National's Microbus. The 86-terminal connector, on 0.156" centers serves as the system bus. Two 50- and one 26-terminal connectors, with pins on 0.1" centers for use with commonly available flat-cable connectors, accommodate the programmable peripheral-I/O and serial-I/O ports. The board also has a 60-terminal connector, with pins on 0.1" centers, adjacent to the system bus, for unassigned signals. Fabricated of blue, FR-4 epoxy glass composite material, the boards are clad with two-ounce, 0.0028"-thick copper conductors. Power buses are solder plated for easy termination while the card-edge connectors are gold-flashed, nickel plated for long life and low resistance.

The 4608 boards, priced at $45 and $34 are available in 1 to 4 quantities off the shelf. For more information contact Vector Electronic Company, Inc., 12460 Gladstone Avenue, Sylmar, CA 91342. *Circle No. 194.*

Heath Announces New Logic Probe for TTL and CMOS Testing

Heath Company recently released IT-7410/ST-7410 **Logic Probes** designed for in-circuit testing of TTL or CMOS circuitry and lamps that turn on when the input voltage crosses the appropriate level. A memory circuit turns on an LED when either threshold level is crossed.

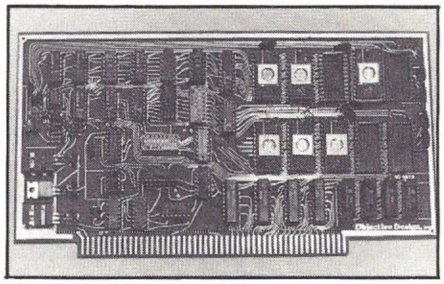
The manufacturer said the new probes provide true logic level detection at high frequencies (not AC-coupled) and that it will detect pulses as short as 10 nS. Upper frequency limits are 100 MHz (TTL or CMOS @ 5 VDC squarewave) and 80 MHz (CMOS @ 15 VDC squarewave). Power for the Logic Probe is drawn from the circuit under test via two spring loaded, insulated clips. A ground lead provides high frequency operation. Probe overload protection is 50 VDC continuous and 175 VDC for 5 seconds. The IT-7410 is the kit version while the ST-7410 is the assembled version. The two are otherwise identical.

For more information about the probes, mail order priced at $39.95 kit and $64.95 assembled, send for your free copy of the latest Heathkit catalog. Write Heath Company, Dept. 350-690, Benton Harbor, MI 49022. *Circle No. 197.*

PROM Programming and Storage Board by Objective Design

A maximum of eight TMS 2716 or 2708 PROMs (16K or 8K bytes) are held on Objective Design's new **PROM programming and storage board**, the Databank. The board will also program PROMs via two special sockets. One of these sockets provides a connection to an external programming station while the other socket allows programming PROMs on the Databank. Each of the eight PROMs may be individually switched into or out of the system address space. In addition, the entire board can be disabled and enabled by IO commands.

Databank will also hold 1K or 2K of 2114 RAM. The RAM will operate as bus memory or can be substituted by software command for any of the PROMs. A PROM in the programming



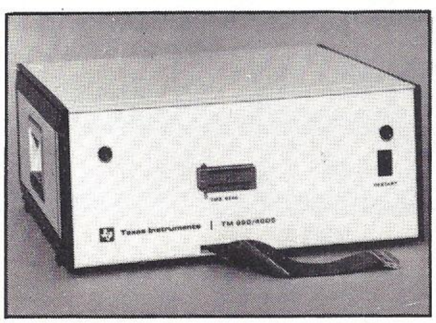
socket also has this substitution ability, permitting a user to copy a PROM set or try a change in the existing program without moving or changing any PROMs. All programming voltages are provided by the Databank board circuitry. The board comes in kit form at the following prices: DB00 (no RAM) $199.95; DB08 (1K RAM) $219.95; DB16 (2K RAM) $239.95 with shipping charges of $5 U.S. and Canada

and $25 overseas. For more information and orders, contact Objective Design, Inc., P.O. Box 20325, Tallahassee, FL 32304, (904) 224-5545. *Circle No. 196.*

Low-Cost Development System by TI for the TMS9940 Microcomputer

A new low-cost **development system** for TI's TMS9940 16-bit single-chip microcomputer is available from Texas Instruments Incorporated. Designated the TMS990/40DS, the system is designed to assemble, edit and link TMS-9940 software, to write the software into the TMS9940 EPROM and to simulate the TMS9940 for system debugging.

As an alternative to traditional front-panel indicators, the TM990/40DS provides an input/output system enabling programmers to use a range of EIA compatible and 20 mA current loop terminals. The heart of the system is a TM990/100M microcomputer board with 1K x 8 bits of RAM and 8K x 8 bits of EPROM. This controls the I/O to the peripherals and to the two emulator boards which provide the functions of the TMS9940 and 2K bytes of RAM to represent the TMS9940 EPROM.



For software development, the TM990/40DS offers a line-by-line assembler tha accepts all of the TMS9940 instructions plus the data and text directives. The user can then implement the TIBUG II monitor to test and debug software prior to programming the EPROM on the TMS9940E.

With TIBUG II, the user can execute the program in his target system through the use of the TISE (Trial In-System Evaluation) cable, a three-foot 40-conductor flat cable assembly with a 40-pin male connector that can be plugged into the intended TMS9940 application socket. TIBUG II also offers a TMS9940E EPROM programmer featuring self-contained control and timing information. The user can program the EPROM from the TM990/40DS RAM, copy the EPROM program into the development system RAM and then verify the contents of the EPROM.

Priced at $2750, the TM990/40DS is available through TI authorized distributors or through Texas Instruments field sales offices. For more information contact Texas Instruments, Inquiry Answering Service, P.O. Box 1443, M/S 653 (Attn: 9940/DS), Houston, TX 77001; or call John Caulfield, (713) 776-6511, Ext. 568. *Circle No. 200.*

New Micronta Digital Frequency Counter

Radio Shack announced its new Micronta **Digital Frequency Counter** for making accurate frequency measurements in audio, RF, video, ultrasonic and digital applications.

Operation is completely automatic. Simply turn on the power switch, connect the input leads to a signal source, or install the mini-rod antenna, and take a reading.

The counter offers a 0.6" six-digit LED display with lead zero blanking. Frequency resolution is 100 Hz throughout the 100 Hz to 45 MHz range, and it is protected against input overvoltage.

The pocket-calculator-sized portable counter may be used to check CB, Ham, business band and radio control transmitters; to establish precise frequencies, bandwidth limits, check exact resonant frequencies and crossovers in audio applications; and to monitor frequency stability of power lines and check operation of pulsing controllers.

It's also suitable for adjusting test equipment output and monitoring, checking computer clock frequencies and operating other digital timebase devices, checking video sync circuits, scanning frequencies and stability testing.

Accuracy is plus or minus 3 ppm at 25 degrees C. Display update is 5 times/second via internal multiplexing. Gate-time is 0.1 second, and size is 6" x 3-1/8" x 1-3/4". The unit requires one 9-volt battery (alkaline recommended) or optional AC adapter.

The Micronta Digital Frequency Counter comes with carry pouch, mini-rod antenna and test leads; price is $99.95. For more information, contact Radio Shack, 1400 One Tandy Center, Fort Worth, TX 76102; (817) 390-3272. *Circle No. 198.*

COMPLEMENTS

Standoffs Aid Component Soldering

Teflon standoffs that facilitate reliable mounting of capacitors and other electronic components are available from Spec Labs, Inc.

Spec Labs Tefon standoffs are precision-cut 0.045" diameter tubes designed to raise capacitors so that ceramic drippings on the leads cannot enter PC holes and interfere with soldering or produce electrical discontinuities. When used with plated through hole boards, the standoffs draw additional solder into the holes by capillary action.

Made of Teflon FEP, Spec Labs standoffs can be used to set any tvpe of component at a precise height above the board surface. Available off the shelf in lengths from 1/16" to 3/8", they can be supplied transparent or color coded by lengths. The manufacturer specializes in #26 to 1/4" diameter flexible tubing cut to tolerances of 0.001".

Spec Labs Teflon standoffs are priced according to quantity and length, with orders shipped in 2 to 5 days. Literature and free samples are available on request. For more information contact Spec Labs, Inc., Edward Abbondanzio, Marketing, 241 Crescen St., Waltham, MA 02154; (617) 891-5200. *Circle No. 204.*

ND 200 Series Line Surge Suppressors

A new ND 200 series of **Line Surge Suppressors** is available from Novadyne Incorporated. These devices provide wide-band protection of AC-powered equipment from destructive, high-energy surges and transients.

Tests show that these surges occur on at least a daily basis on all industrial and residential power distribution systems. On a typical 120 VAC line, surges generated by sources external to the

building may run as high as 6000 volts. Switching of internal loads is generally responsible for repetitive transients of up to 2500 volts.

The effects of these surges on electrical and electronic equipment range from annoying disturbances through gross malfunctions to catastrophic failure. Equipment life is reduced by repetitive overstressing of insulation and/or components. Especially susceptible are computers and peripherals, color TV and stereos, communications, copiers and calculators, medical electronics and other industrial and residential electrical and electronic equipment.

The ND 200 Line Surge Suppressor protects such equipment by absorbing excess transient energy with a state-of-the-art suppression device. A supplemental ferrite filter reduces spikes and transients which fall below the level of the suppressor.

Models are available for 120 or 240 V, 20 amp, single phase and for 240 V, 20 or 30 amp, single or three phase.

For literature and pricing write, Novadyne Incorporated, 11702 Trask Avenue, Garden Grove, CA 92643; (714) 636-4620. *Circle No. 206.*

TI Introduces LCD Analog Watch

Texas Instruments Incorporated announced its totally **electronic LCD quartz analog watch**, The Time Indicator. The watch displays time with electronic hands via a movement with no moving parts.

TI research indicates that some people can tell time more easily when using a watch with hands, because of the spatial relationship of the hands and the dial face. They know certain positions mean certain events are supposed to occur and are uncomfortable with digital watches because they can't see the time relationship visually. Mechanical watches, though, have movements with gears and wheels that wear out and must be cleaned periodically. In addition, they cannot inexpensively offer stopwatch combinations, dual time zone indicators, automatic calendars, and a number of other special functions that are usually available only on watches with digital time displays.

Instead of hands and gears, TI's watch uses liquid crystal indicators that sweep the dial face in a manner closely akin to mechanical hands, driven by a microelectronic module. Timekeeping displays at the user's command include hours and minutes in another time zone; day and date; and a stopwatch giving elapsed time in hours and minutes; minutes and seconds; and seconds and tenths of seconds. Accuracy is within 15 seconds a month.

Suggested retail prices for the model range from $275 to $325. Available through jewelry and department stores, the watches are anti-magnetic and shock resistant.

For more information contact Texas Instruments Inc., Consumer Relations, P.O. Box 53 (Attn: Time Indicator), Lubbock, TX 79408. *Circle No. 202.*

CLASSIFIEDS

Cassette Duplication Services Offered by Microsette Company

Microsette Co., manufacturer of short length cassettes for home and hobby computers, announces **cassette program duplication services** for TRS-80 and PET computers.

Microsette provides duplication of cassette programs in any volume from 100 to 100,000. Each cassette is backed backed with a 60-day warranty. In addition, duplicated cassettes may be made exact length per program with no excess tape to consume rewind time.

Prices start at $1 each, including the cassette, a Norelco-style box, unaffixed blank labels for your use and shipping of the finished product to a single point anywhere in the USA.

For details contact Microsette Co., 777 Palomar Ave., Sunnyvale, CA 94086. *Circle No. 208.*

Leader Short Form Catalog Features Advanced Design Test Instruments

An enlarged group of **test and measuring instruments** is featured in a new short-form color catalog from Leader Instruments Corp. The 12-page booklet describes eight oscilloscope models with bandwidths from 4 to 30 MHz; eight professional audio instruments including two audio system analyzers; four multimeters including digital and analog versions; 80 and 250 MHz frequency counters; three color bar generators including an NTSC Model; an FM multiplex generator, LCR bridge and other general purpose instruments and accessories.

All Leader instruments carry a two year limited warranty backed by factory service centers on the East and West Coasts. For more information contact Leader Instrument Corp., 151 Dupont St., Plainview, NY 11803; (516) 822-9300. *Circle No. 217.*

Computer Data Directory Provides Comprehensive Index

The **Computer Data Directory** announces its comprehensive index of small computer products companies. 675 companies. Included are brand name manufacturers of systems, peipherals and accessories. In software, you can find who handles languages, business applications, household control, games – even custom programming services. Miscellaneous information includes where to find books, magazines, newsletters, home study courses, data banks, tools and repair services. Computer stores are indexed geographically. The book includes a reader service card to all listed companies.

This reference source, over 100 pages, costs $4.98, postage included. For more information contact The Computer Data Directory, P.O. Box 598, Cleveland, OH 44107. *Circle No. 211.*

ITC Introduces Verbatim Mini-disk

Information Terminals Corp., manufacturer of diskettes and removable magnetic storage media, announced the production of a new **miniaturized flexible disk.**

Using floppy disk media formulations, the minidiskette, dubbed the MD 525, is about half the size of standard floppy disks and provides one-third the storage capacity. The MD

525 can store the equivalent of 25 typed pages of data, and is compatible with the new small disk drive systems like those manufactured by Shugart Associates. The MD 525 is functional in such applications as power typing systems, editing systems, word processing and microcomputer hobby uses, the company said.

Price of the ITC MD 525, which coincidentally is 5.25" in diameter, is $5.25, and delivery is from ITC distributor stocks. For more information contact ITC, 323 Soquel Way, Sunnyvale, CA 94086. *Circle No. 210.*

New Data Sheet Describes Millennium's Microsystem Analyzer

Millennium Systems, Inc., has published a new data sheet describing its **MicroSystem Analyzer**, which supports a variety of microprocessors. This publication may interest people involved in the design, production-test and field service of microprocessor-based products. Weighing 21 pounds, the MicroSystem Analyzer is small enough to fit under an airline seat.

The MicroSystem Analyzer provides both in-circuit emulation and signature analysis; it delivers both test and fault diagnostic capabilities to the user.

The six-page data sheet explains in-circuit emulation and signature analysis as well as the time-domain measurement capability of the test instrument. It also describes how the unit works in design, production test and field service applications. An illustration identifies operating displays and pushbuttons and a data section supplies both general operational and fault detection capabilities.

For a copy of the data sheet, entitled "Millennium μSA MicroSystem Analyzer", contact Marty Weisberg, Millennium Systems, Inc., 19020 Pruneridge Avenue, Cupertino, CA 95104; (408) 996-9109. *Circle No. 215.*

Oxide Free Solder Cream

Multicore Solders has developed a processing technique for the manufacture of its line of **Solder Creams**, resulting in an oxide-free material. This achievement will eliminate solder balls, joint contamination and dewetting while substantially improving the performance and shelf-life of the material, the company said.

Because of this new manufacturing method, Multicore Solder Creams require less heat, shorter dwell time and lowe soldering temperatures, according to the company. The flux residue is light and clear in color giving it a cosmetic look for an easily inspected joint. Wetting properties are improved since flux activity is not expended in removing oxides from the metal powder during heating but rather is concentrated on the surfaces to be soldered.

Multicore Solder Cream is available in a variety of tin-lead as well as silver bearing alloys with either an active or mildly active rosin base flux. Purity, mesh size and flux properties conform to QQ-S-571. Stirring is normally not required prior to use. Its tacky consistency enables it to be used as a temporary adhesive to hold component parts in place prior and during application of heat.

The material is useful for thick and thin film circuits, is printable and flexibile, is an effective and economic substitute for many preforms. Depending on the application and method of transferring the cream to the surface to be soldered, it is available either in 500 gm jars for screening, stencilling, brushing and automatic dispensing or in a 25 gm refillable (or disposable) hand operated

ADVERTISERS' INDEX

SALES OFFICES

Northeast: George Palken, 1050 Commonwealth Ave., Boston, MA 02215; (617) 232-5470 • **Mid-Atlantic**: Arthur Daks, 104 So. Central Ave., Valley Stream, NY 11580; (516) 872-9550 • **Midwest**: Hank Bean, 2633 Hillside Lane, Evanston, IL 60201; (312) 475-7173 • **Los Angeles**: Carol Stagg, 924 Westwood Blvd., Suite 610, Los Angeles, CA 90024; (213) 487-3017 • **San Francisco**: Ralph Petersen, 1180 Oak Creek Dr., Palo Alto, CA 94304; (415) 328-4392 • Japan: Hiro H. Irie, International Business Corp., 11-8, Narita-Higashi 1-chome, Suginami-KU Tokyo 166; Phone: (03) 311-1746 • Exposition Service Sales Office: Louise Garcia, (505) 898-4562.

syringe for use as a "liquid preform". Excellent fine line definition can be easily achieved eliminating "solder bleeding" caused by oxides incorporated into the formulation during manufacture. As a liquid preform it can be deposited with pin point accuracy in any configuration and in controlled quantities.

Depending on alloy and quantity, prices average about ten cents ($.10) per gm, available from stock.

Complete information and sample evaluation kits are available from Multicore Solders, Westbury, NY 11590. *Circle No. 203.*

"Smart Light Switch" for Burglary Prevention

Vigilite, Hutec Corporation's microprocessor-based **pre-programmed light control** fits into a home wall switch box, replacing the conventional on/off switch with a "smart" light switch. The same microprocessor provides a digital clock.

Easily installed, Vigilite begins immediately its dual function as wall clock and burglar deterrent. The user selects from pre-programmed independent lighting patterns for bedroom, bathroom, kitchen, living room or outside porch lights. Two to three Vigilite units in the house will simulate normal lighting patterns that discourage burglars. An algorithm built into the microprocessor's programming controls lights in a realistic random pattern. Because it's pre-programmed, the user has only to push a button to activate Vigilite.

Vigilite costs $39.95. For more information contact Hutec Corporation, 1050 East Duane, Sunnyvale, CA 94086. *Circle No. 205.*

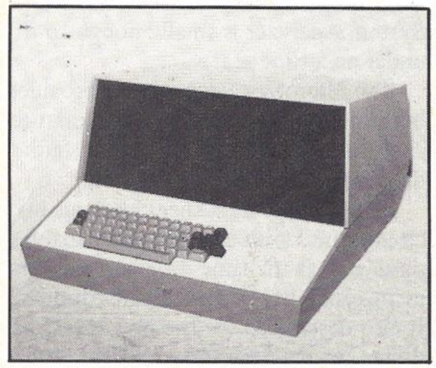
The MOS Saver Service Kit

Micro Electronic Systems, Inc., offers a new **field service kit** for prevention of static charges during repair operations on PC boards. Called the MOS Saver, p/n 1420, the kit consists of a 18" x 24" conductive work mat, a conductive wrist strap and an 8" conductive ground strap.

Price is $17.50 in single quantities. For more information contact Micro Electronics Systems, 159 Main St., Danbury, CT 06810; (203) 797-1441. *Circle No. 212*

Plastic CRT Terminal and Keyboard Enclosures

CEI has added CRT **terminals and keyboard enclosures** to their product line. Three models are available; CE20, 21 and 22 accommodate different size CRTs and styles. Model CE10 is the standard keyboard enclosure. All units are molded of high impact Noryl ther-



moplastic and are available either painted or unpainted.

CRT units have plexiglass screens, ventilation grills and blower openings. All units feature integral component mounting bosses and removeable keyboard mounting plates. The CRT units are comprised of three molded parts: base, keyboard and shroud. Units range in price from $20.95 to $83.95. For more information contact Bob Thomas, Custom Electronics Industries, 609 Route 109, W. Babylon, NY 11704; (516) 884-2121. *Circle No. 209.*

Short Form Catalog on Data Communications Devices

A new illustrated **short-form catalog** has been released by Gandalf Data, Inc. The catalog covers short and limited distance modems, business machine interfaces, test sets for installation and maintenance and Private Automatic Computer Exchange (PACX) Systems. Synchronous and Asynchronous modems are available and transmit data up to 19,200 bps out to 50 miles. The units are sold individually or can be rack mounted for expanded operations. PACX is an alternative to the switched telephone network for terminal switching, port selection and port contention. For more information, request SFC 78 from Gandalf Data, Inc., 1019 S. Noel, Wheeling, IL 60090. *Circle No. 216.*

Vaco Announces Its Combination Crimping Tool

Combination Crimping Tool No. 1963 was recently introduced by Vaco. The No. 1963 crimps both insulated and non-insulated terminals. The wire cutter is at the tip of the tool to facilitate cutting in cramped quarters. It strips 22-10 gauge cable, with stripping holes numbered according to wire gauge size to make stripping easier. The combination crimping tool also slices six popular bolt sizes – 4-40, 5-40, 6-32, 8-32, 10-24 and 10-32. Just screw the bolt into the proper size threaded hole, place to desired length and squeeze. The bolt is cleanly sliced and leaves no burrs to file. Cushioned grip handles ease operation. The entire tool has a black oxide finish.

For more information contact Vaco Products Company, 1510 Skokie Blvd., Northbrook, IL 60062; (312) 564-3300.
Circle No. 213

Catalog on 6800 Development Software

Descriptions of **6800 development software**, including STRUBAL, a new high-level language compiler for 6800s, are available in Hemenway Associates, 1978 SOFTWARE CATALOG. The features of each software product are listed in table form for reference. In addition, listings of assembler directives, operators, options and EDIT commands are given. For more information contact Hemenway Associates, Inc., 151 Tremont St., Suite 8P, Boston, MA 02146; (617) 426-2641.
Circle No. 214

Santa Claus is coming to PC

Next month, as a special holiday offering, we'll focus our What's Coming Up product section on gift ideas. So treat yourself, your friends and your computer to something special for Christmas, using our handy listings as a shopping guide.

As an added bonus – our holiday gift to you – the December issue will include a comprehensive index to articles appearing in *PC* during the last two years.

Two Gifts in One

Start a new subscription to *Personal Computing* now, for yourself and/or your friends, or renew/extend your current subscription, and you'll benefit two big ways. First, you'll receive each monthly issue of *Personal Computing,* the consumer magazine packed with business and home programs, programming techniques, buying guides, product news and other consumer information to help make your computer a powerful tool for business and home.

Second, act before January first and you'll receive these bonuses:

• Start a new subscription with the January issue, and you'll receive free the valuable December issue containing a Christmas buying guide and an index to articles appearing in *Personal Computing* in the last two years. • Buy more than one subscription (they make great gifts) and you'll pay $2 less per year for each subscription after the first one. *(And each extra subscription receives the great December issue as a bonus also.)* • Renew your current subscription before January first, and you'll get a free bonus issue added to your subscription. • Prepay your subscription or renewal/extension, and you'll get *another* bonus issue per year added to your subscription. (That's two free issues for a one-year subscription; three free issues for a two-year subscription; and four for a three-year subscription.) Prepay your gift subscriptions and each subscription receives the bonus issues. • For each gift subscription, we'll send a gift card in your name. Just check the box on the subscription order coupon.

So act now, using the hand coupon provided, *and* receive the free bonus issue.

FIRST SUBSCRIPTION – USA
☐ 1 year $14
☐ 2 years $26
☐ 3 years $38

ADDITIONAL SUBSCRIPTIONS – USA
☐ 1 year $12 (Save $2 per subscription)
☐ 2 years $22 (Save $4 per subscription)
☐ 3 years $32 (Save $6 per subscription)

ADDITIONAL POSTAGE – per subscription per year: Canada and Mexico – $4 surface, $8 air; Other Foreign – $8 surface, $36 air. Please remit US funds.

☐ Check enclosed *(and receive one extra issue for each year)*

Charge my ☐ Master Charge ☐ Visa

Account # ____________ Expires ________

Please ☐ start ☐ renew/extend my subscription to

Personal Computing

Name ____________

Address ____________

City ________ State ________ Zip ________

New subscriptions receive bonus December issue; renewals/extensions receive additional issue added to subscription.

☐ Also, send gift subscriptions in my name to:

Name ____________

Address ____________

City ________ State ________ Zip ________

Name ____________

Address ____________

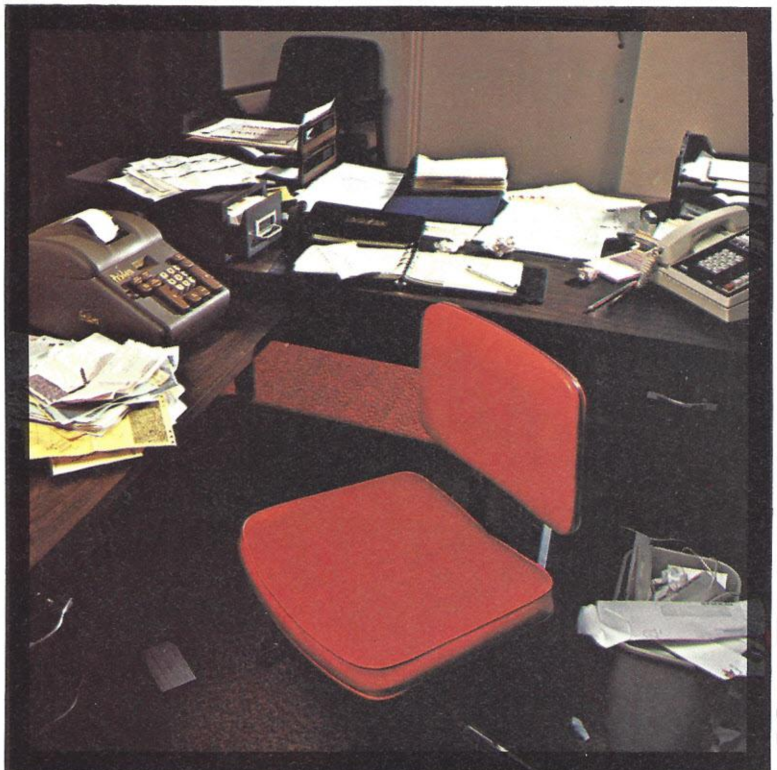
City ________ State ________ Zip ________

☐ Please send gift cards in my name.

To order more than two gift subscriptions, please make copies of this card. We cannot bill for gift subscriptions.

MAIL TO: **Personal Computing** 1050 Commonwealth Ave., Boston, MA 02215.

Is your TRASH FLOW higher than your CASH FLOW?



SOFTWARE LIBRARY

Five diskettes are included to give you immediate operating and programming capabilities.

DISK 1 is a master diskette with BASIC, MDOS, Text Editor, Assembler and more!

DISK 2 has many games including LUNAR, CRAPS, and SPLAT. There's room left for you to add your own.

DISK 3 contains a Small Business Accounting package which includes Accounts Payable and Receivable, Inventory, a General Ledger, and more.

DISKS 4 & 5 are blank so you may add your own programs.

Clean it up with a VERSATILE 3B T.M.

If you're a businessman, we know how difficult it is to keep neat and efficient accounting records. Let our **VERSATILE** systems do it for you. You'll have a complete system built into a single cabinet, and a free software library on diskette to get your computer working for you the first day. $3295 Assembled and tested.

WHAT ELSE DOES A COMPLETE SYSTEM NEED?

Contact Our Distributors for Regional Sales and Service

Alexander & Co., Inc.
5518 Florin Road
Sacramento, CA 95823
(916) 422-9070

Huron Electronics
415 N. Silver
Bad Axe, Mich. 48413
(517) 269-9267

The Computer Store
3801 Kirby Dr., Suite 432
Houston, Texas 77098
(713) 522-7845

Southeast Representative:

Scientific Sales Co.
175 W. Wieuca, Suite 210
Atlanta, GA 30342
(404) 252-6808

DealerApplications Available.

COMPUTER DATA SYSTEMS

5460 Fairmont Drive • Wilmington, Delaware 19808 • 302-738-0933

CIRCLE 2

COSMAC VIP

$249* gets the entire family into creating video games, graphics and control functions. For starters.

COSMAC VIP, the completely assembled, ready-to-operate RCA Video Interface Processor, opens up a whole new world of computer excitement. New challenges in graphics, games and control functions. Yet it's just $249.00.

Easy to buy. And easy to program, thanks to its unique, easy-to-use interpretive language. You get a complete how-to book including programs for 20 games: fun, challenging, and ready to load and record on your cassette.

Simple but powerful.

Built around an RCA COSMAC microprocessor, the VIP is a complete computer system that can grow with you. It has 2K of RAM, expandable on-board to 4K. Plus a ROM monitor, audio tone output to a built-in speaker, power supply, and 8-bit input and output ports for control of relays, sensors, or other peripherals.

Soon RCA will offer options for color graphics and 256 tone sound generation. An optional auxiliary keyboard will open up an exciting world of two-player games.

Take the first step now.

Check your local computer store or electronics distributor for the VIP. Or contact RCA VIP Marketing, New Holland Avenue, Lancaster, PA 17604. Phone (717) 291-5848.

*Suggested retail price. Does not include video monitor or cassette recorder.

The fun way into computers.

RCA

CIRCLE 1